Framework

Framework AWS Well-Architected

Framework AWS Well-Architected: Framework

Table of Contents

Framework AWS Well-Architected

Data di pubblicazione: 6 novembre 2024 (Revisioni del documento)

Il Framework AWS Well-Architected aiuta a comprendere i pro e i contro delle decisioni prese durante la progettazione di sistemi in AWS. Utilizzando il Framework, scoprirai le best practice architetturali per progettare e gestire sistemi affidabili, sicuri, efficienti, convenienti e sostenibili nel cloud.

Introduzione

Il Framework AWS Well-Architected aiuta a comprendere i pro e i contro delle decisioni prese durante la progettazione di sistemi in AWS. Utilizzando il Framework, scoprirai le best practice architetturali per progettare e gestire carichi di lavoro affidabili, sicuri, efficienti, convenienti e sostenibili nell'Cloud AWS. Offre un metodo per misurare con coerenza le proprie architetture rispetto alle best practice e individuare le aree di miglioramento. Il processo di revisione di un'architettura è una conversazione costruttiva sulle decisioni relative all'architettura e non un meccanismo di audit. Disporre di sistemi ben architettati aumenta notevolmente la probabilità di successo aziendale.

Gli AWS Solutions Architect vantano anni di esperienza nell'architettura di soluzioni in un'ampia gamma di business e casi di utilizzo. Abbiamo supportato migliaia di clienti nella progettazione e revisione delle loro architetture su AWS. Grazie a questa esperienza, abbiamo identificato best practice e strategie principali per i sistemi di architettura nel cloud.

Il Framework AWS Well-Architected documenta un insieme di domande fondamentali per capire se un'architettura specifica si allinea bene con le best practice del cloud. Il framework fornisce un approccio coerente per la valutazione dei sistemi rispetto alle qualità che ti aspetti da sistemi basati sul cloud moderni e i rimedi necessari per raggiungere tali qualità. Man mano che AWS continua a evolversi e noi continuiamo a imparare di più dal lavoro che svolgiamo con i nostri clienti, continueremo a ridefinire la definizione di architettura ottimale.

Questo framework è rivolto a chi svolge ruoli tecnologici, ad esempio ai Chief Technology Officer (CTO), ai progettisti, agli sviluppatori e ai membri dei team operativi. Descrive le best practice e le strategie AWS da usare per la progettazione e il funzionamento di un carico di lavoro cloud, e fornisce collegamenti a ulteriori dettagli di implementazione e pattern architetturali. Per ulteriori informazioni, consulta la homepage di AWS Well-Architected.

AWS offre anche un servizio gratuito di revisione dei carichi di lavoro. AWS Well-Architected Tool (Strumento AWS WA) è un servizio cloud che fornisce un approccio coerente per la revisione e la

valutazione della tua architettura tramite il Framework AWS Well-Architected. Lo Strumento AWS WA fornisce suggerimenti per rendere i carichi di lavoro più affidabili, sicuri, efficienti e convenienti.

Per aiutarti ad applicare le best practice, abbiamo creato AWS Well-Architected Labs, che fornisce un repository di codice e documentazione per un'esperienza concreta di implementazione delle best practice. Abbiamo anche collaborato con partner della rete dei partner AWS selezionati, che sono membri del programma per partner AWS Well-Architected. Tali partner AWS vantano una conoscenza approfondita di AWS e possono aiutarti nella revisione e nel miglioramento dei tuoi carichi di lavoro.

Definizioni

Tutti i giorni, gli esperti AWS supportano i clienti nella progettazione di sistemi di architettura per sfruttare le best practice nel cloud. Ti aiutiamo a trovare i compromessi relativi all'architettura nel processo di evoluzione dei tuoi progetti. Quando implementi questi sistemi in ambienti live, analizziamo le prestazioni di questi sistemi e le conseguenze dei suddetti compromessi.

Sulla base di quello che abbiamo imparato, abbiamo creato il Framework AWS Well-Architected, che fornisce a clienti e partner un insieme coerente di best practice per valutare le architetture, e comprende un insieme di domande che puoi utilizzare per valutare se la tua architettura è ben allineata alle best practice AWS.

Il Framework AWS Well-Architected si basa su sei pilastri: eccellenza operativa, sicurezza, affidabilità, efficienza delle prestazioni, ottimizzazione dei costi e sostenibilità.

Tabella 1. I pilastri del Framework AWS Well-Architected

Nome	Descrizione
Eccellenza operativa	Comprende la capacità di supportare lo sviluppo ed eseguire carichi di lavoro in modo efficace, ottenere informazioni approfondite sulle loro operazioni e migliorare continuam ente i processi e le procedure di supporto per offrire valore aggiunto.
Sicurezza	Il pilastro della sicurezza descrive come sfruttare le tecnologie cloud per proteggere

Nome	Descrizione
	dati, sistemi e risorse in modo da migliorare il livello di sicurezza.
Affidabilità	Il pilastro dell'affidabilità comprende la capacità di un carico di lavoro di eseguire la funzione attesa in modo corretto e coerente quando previsto. Ciò comprende la possibilità di utilizzare e testare il carico di lavoro per tutto il ciclo di vita. Il presente documento fornisce linee guida dettagliate sulle best practice per l'implementazione di carichi di lavoro affidabili in AWS.
Efficienza delle prestazioni	La capacità di utilizzare in modo efficiente le risorse di elaborazione per soddisfare i requisiti di sistema e di mantenere tale efficienza di fronte al cambiamento delle richieste e all'evoluzione delle tecnologie.
Ottimizzazione dei costi	La capacità di eseguire i sistemi per distribuire il valore aziendale al prezzo minore.
Sostenibilità	La capacità di migliorare continuamente l'impatto sulla sostenibilità riducendo il consumo energetico e aumentando l'efficienza di tutti i componenti di un carico di lavoro, massimizzando i benefici delle risorse allocate e riducendo al minimo le risorse totali richieste.

Nel Framework AWS Well-Architected, si utilizzano i seguenti termini:

- Un componente è il codice, la configurazione e le risorse AWS che insieme soddisfano un requisito. Spesso un componente è l'unità di proprietà tecnica ed è disaccoppiato da altri componenti.

- Con il termine carico di lavoro ci riferiamo all'insieme di componenti che forniscono valore aziendale. Un carico di lavoro, normalmente, è il livello di dettaglio comunicato dai leader aziendali e della tecnologia.
- Secondo il nostro punto di vista, l'architettura è il modo in cui i componenti interagiscono in un carico di lavoro. Il modo di comunicare e di interagire dei componenti è spesso l'aspetto principale dei diagrammi architetturali.
- Le tappe fondamentali indicano cambiamenti chiave della tua architettura man mano che si evolve nel corso del ciclo di vita del prodotto (progettazione, test, messa online e produzione).
- Nell'ambito di un'organizzazione il portfolio delle tecnologie rappresenta l'insieme di carichi di lavoro necessari affinché l'azienda possa essere operativa.
- Il livello di impegno è la categorizzazione della quantità di tempo, sforzo e complessità che un'attività richiede per la sua realizzazione. Ogni organizzazione deve considerare le dimensioni e le competenze del team e la complessità del carico di lavoro per ottenere un contesto aggiuntivo che consenta di classificare correttamente il livello di impegno.
  - Elevato: il lavoro potrebbe richiedere più settimane o più mesi. Potrebbe essere suddiviso in molteplici fasi, rilasci e attività.
  - Medio: il lavoro potrebbe richiedere più giorni o settimane. Potrebbe essere suddiviso in molteplici rilasci e attività.
  - Basso: il lavoro potrebbe richiedere più ore o giorni. Potrebbe essere suddiviso in molteplici attività.

Quando progetti l'architettura dei carichi di lavoro, devi trovare dei compromessi tra i pilastri su cui si regge il tuo contesto aziendale. Le decisioni aziendali possono stabilire le priorità di progettazione. Potresti ottimizzare per migliorare la sostenibilità e ridurre i costi a spese dell'affidabilità in ambienti di sviluppo oppure, per quanto riguarda le soluzioni mission-critical, potresti ottimizzare l'affidabilità a fronte di costi più elevati e di un impatto ambientale maggiore. Nelle soluzioni di e-commerce, le prestazioni possono avere un impatto sui profitti e sulla propensione all'acquisto da parte dei clienti. Solitamente, la sicurezza e l'eccellenza operativa non sono soggette a compromessi rispetto agli altri pilastri.

Architettura

Negli ambienti on-premises, i clienti spesso hanno un team centrale per l'architettura delle tecnologie che funziona da livello superiore per altri team di prodotto o funzionalità, al fine di garantire che i team rispettino le best practice. I team dell'architettura delle tecnologie spesso sono composti da diversi

ruoli come il Technical Architect (infrastruttura), il Solutions Architect (software), il Data Architect, il Networking Architect e il Security Architect. Spesso questi team utilizzano TOGAF o il Framework Zachman nell'ambito di una funzionalità dell'architettura aziendale.

Noi di AWS preferiamo distribuire le competenze tra i team, invece di centralizzarle in un unico team. Quando si sceglie di distribuire il potere decisionale si corrono dei rischi, ad esempio il rischio di garantire che i team interni rispettino gli standard. Noi mitighiamo questo rischi in due modi. In primo luogo, disponiamo di pratiche (modalità per eseguire attività, processi, standard e norme accettate) che hanno lo scopo di permettere a ogni team di possedere tali competenze e ci serviamo di esperti che verificano che i team adottino standard più severi di quelli che devono rispettare. In secondo luogo, implementiamo meccanismi che eseguono controlli automatizzati per verificare che gli standard vengano rispettati.

> "Le buone intenzioni non bastano mai, per avere successo servono buoni meccanismi", Jeff Bezos.

Questo significa sostituire gli sforzi di una persona con meccanismi (spesso automatizzati) che verificano la conformità alle regole e ai processi. Tale approccio distribuito è supportato dai principi di leadership di Amazon e stabilisce una cultura in tutti i ruoli che parte dal cliente. Il lavoro a ritroso è una parte fondamentale del nostro processo di innovazione. Partiamo dal cliente e da quello che vuole e sulla base di questo definiamo e indirizziamo i nostri sforzi. I team che mettono il cliente al centro sviluppano prodotti sulla base delle necessità del cliente.

Per l'architettura questo significa che ci aspettiamo che ogni team sia in grado di creare architetture e di seguire le best practice. Per aiutare i nuovi team ad acquisire queste competenze o i team esistenti ad alzare il livello, abilitiamo l'accesso a una community virtuale di capo ingegneri che possono eseguire la revisione dei loro progetti e aiutarli a comprendere le best practice di AWS. La community di capo ingegneri lavora per rendere visibili e accessibili le best practice. Uno dei modi per fare ciò, ad esempio, è servirsi delle lunchtime talk che si concentrano sull'applicazione di best practice a esempi reali. Le lunchtime talk sono registrate e possono essere utilizzate come materiale di onboarding per i nuovi membri del team.

Le best practice AWS sono il risultato della nostra esperienza nell'esecuzione di migliaia di sistemi su Internet. Preferiamo utilizzare i dati per definire le best practice, ma ci serviamo anche di esperti in materia, come i capo ingegneri. Quando i capo ingegneri vedono emergere nuove best practice, lavorano con la community per verificare che i team le rispettino. Con il tempo, queste best practice

vengono formalizzate nei nostri processi di revisione interna e nei meccanismi che rafforzano la compliance. Il Framework Well-Architected è l'implementazione del nostro processo di revisione interno rivolta ai clienti, in cui abbiamo codificato la nostra idea di ingegneria responsabile attraverso ruoli di campo come Solutions Architect e i team di ingegneria interni. Il Framework Well-Architected è un meccanismo scalabile che consente di trarre vantaggio da questi insegnamenti.

Seguendo l'approccio della community di capi ingegneri con la proprietà distribuita dell'architettura, riteniamo che si possa ottenere un'architettura aziendale Well-Architected che si basa sulle necessità del cliente. I leader della tecnologia (come i CTO o i manager dello sviluppo) che eseguono revisioni Well-Architected tra tutti i carichi di lavoro ti permettono di comprendere più a fondo i rischi relativi al portfolio delle tecnologie. Tramite questo approccio puoi identificare dei temi tra i team che la tua organizzazione può affrontare tramite meccanismi, formazione o dialoghi informali in cui i capo ingegneri possono condividere le loro idee su aree specifiche con diversi team.

Principi generali di progettazione

Il Framework Well-Architected identifica una serie di principi generali per facilitare la corretta progettazione nel cloud:

- Smetti di ipotizzare quali siano le tue esigenze di capacità: se prendi una decisione sbagliata sulla capacità al momento dell'implementazione di un carico di lavoro, rischi di ritrovarti con risorse inattive o ad affrontare le conseguenze della capacità limitata. Con il cloud computing, questi problemi vengono risolti. Puoi utilizzare la capacità di cui hai bisogno e ridurre orizzontalmente o aumentare orizzontalmente il sistema automaticamente.
- Esegui test dei sistemi su scala produttiva: nel cloud, puoi creare un ambiente di test su scala produttiva on demand, completare i test e disattivare le risorse. Poiché paghi per l'ambiente di test solo quando è in esecuzione, puoi simulare un ambiente live a un costo notevolmente inferiore rispetti ai test on-premises.
- Automatizza pensando alla sperimentazione architettonica: l'automazione ti permette di creare e replicare i tuoi carichi di lavoro a basso costo e di evitare le spese della gestione manuale. Puoi tenere traccia delle modifiche all'automazione, effettuare l'audit dell'impatto e tornare ai parametri precedenti, se necessario.
- Considera le architetture evoluzionistiche: in un ambiente tradizionale, le decisioni relative all'architettura spesso sono implementate come eventi singoli e statici, con poche versioni principali di un sistema durante il ciclo di vita. Alla luce del continuo cambiamento di un'azienda e del suo contesto, le decisioni iniziali potrebbero ostacolare la capacità del sistema di soddisfare i requisiti aziendali in evoluzione. All'interno del cloud, la capacità di automatizzare e testare on demand

diminuisce il rischio di impatto dovuto alle modifiche della progettazione. Questo permette ai sistemi di evolversi nel tempo, in modo che le aziende possano trarre vantaggio dalle innovazioni come pratica standard.

- Promuovi le architetture servendoti dei dati: nel cloud puoi raccogliere dati relativi all'impatto delle tue scelte architetturali sul comportamento del tuo carico di lavoro. Questo ti permette di prendere decisioni basate sui fatti su come migliorare il carico di lavoro. La tua infrastruttura cloud è un codice, quindi, puoi usare tali dati a vantaggio delle scelte e dei miglioramenti relativi all'architettura nel tempo.
- Migliora con le giornate di gioco: testa le prestazioni dell'architettura e dei processi pianificando regolarmente giornate di gioco per simulare eventi della produzione. Questi ti aiuta a capire dove puoi apportare dei miglioramenti e ti può aiutare a sviluppare un'esperienza organizzativa nella gestione degli eventi.

I pilastri del framework

La creazione di un sistema software è molto simile alla costruzione di un edificio. Se le fondamenta non sono solide, possono emergere problemi strutturali che minano l'integrità e la funzionalità dell'edificio. Se nella creazione dell'architettura per soluzioni tecnologiche trascuri i sei pilastri di eccellenza operativa, sicurezza, affidabilità, efficienza delle prestazioni, ottimizzazione dei costi e sostenibilità, può diventare complicato sviluppare un sistema che soddisfi le tue aspettative e i tuoi requisiti. L'aggiunta di questi pilastri alla tua architettura ti aiuterà a produrre sistemi efficienti e stabili. Questo ti permetterà di concentrarti su altri aspetti della progettazione, come i requisiti funzionali.

Pilastri

- Eccellenza operativa
- Sicurezza
- Affidabilità
- Efficienza delle prestazioni
- Ottimizzazione dei costi
- Sostenibilità

Eccellenza operativa

L'eccellenza operativa è un impegno a sviluppare correttamente il software garantendo costantemente un'esperienza cliente di alto livello. Il pilastro dell'eccellenza operativa contiene best practice per organizzare il team, progettare il carico di lavoro, farlo funzionare su scala e seguire la sua evoluzione nel tempo.

Il pilastro dell'eccellenza operativa offre una panoramica dei principi di progettazione, delle best practice e delle domande. Puoi trovare linee guida prescrittive sull'implementazione nel whitepaper sul pilastro dell'eccellenza operativa.

Argomenti

- Principi di progettazione
- Definizione
- Best practice
- Risorse

Principi di progettazione

Ecco i principi di progettazione per l'eccellenza operativa nel cloud:

- Organizza i team in base ai risultati aziendali: la capacità di un team di conseguire i risultati aziendali deriva dalla visione della leadership, dall'efficacia delle operazioni e dall'allineamento del modello operativo all'azienda. È necessario che la leadership sia totalmente coinvolta e impegnata nella trasformazione delle operazioni nel cloud (CloudOps) con un modello operativo cloud adeguato che incentivi i team a operare nel modo più efficiente per raggiungere i risultati aziendali. Il modello operativo corretto include persone, processi e capacità tecnologiche per scalare, ottimizzare la produttività e favorire la differenziazione tramite l'agilità, la reattività e l'adattamento. La visione a lungo termine dell'organizzazione si traduce in obiettivi che vengono comunicati alle parti interessate dell'azienda e agli utenti dei tuoi servizi cloud. Gli obiettivi e i KPI operativi sono allineati a tutti i livelli. Questa procedura promuove il valore a lungo termine derivante dall'implementazione dei seguenti principi di progettazione.
- Implementa l'osservabilità per approfondimenti utilizzabili: acquisisci una comprensione completa del comportamento, delle prestazioni, dell'affidabilità, dei costi e dello stato del carico di lavoro. Stabilisci indicatori chiave delle prestazioni (KPI) e usa la telemetria dell'osservabilità per prendere decisioni informate e agire tempestivamente quando i risultati aziendali sono a rischio. Migliora in modo proattivo le prestazioni, l'affidabilità e i costi sulla base di dati sull'osservabilità fruibili.
- Automatizza in modo sicuro, laddove possibile: nel cloud, ti è possibile applicare la medesima disciplina di progettazione che utilizzi per il codice dell'applicazione a tutto il tuo ambiente. Definisci l'intero carico di lavoro e le relative operazioni (applicazioni, infrastruttura, configurazione e procedure) come codice e aggiornarlo. Quindi, automatizza le operazioni del carico di lavoro avviandole in risposta agli eventi. Nel cloud, utilizzi la sicurezza dell'automazione configurando i guardrail, tra cui il controllo della frequenza, le soglie di errore e le approvazioni. Un'automazione efficiente offre risposte coerenti agli eventi, limita l'errore umano e riduce l'impegno degli operatori.
- Applica modifiche frequenti, minime e reversibili: progetta carichi di lavoro scalabili e con accoppiamento debole per consentire l'aggiornamento regolare dei componenti. Le tecniche di implementazione automatizzate insieme a modifiche incrementali più piccole riducono il raggio di esplosione, ovvero l'entità dell'impatto, e consentono un'inversione più rapida in caso di guasti. Ciò aumenta la fiducia necessaria per apportare modifiche strategiche al carico di lavoro mantenendo la qualità e adattandosi rapidamente ai cambiamenti delle condizioni di mercato.
- Perfeziona con frequenza le procedure operative: l'evoluzione delle operazioni deve seguire quella dei carichi di lavoro. Se usi procedure operative, cerca delle opportunità per migliorarle. Organizza regolari revisioni per accertarti che tutte le procedure siano efficaci e che i team le conoscano

adeguatamente. Se vengono individuate delle lacune, aggiorna le procedure di conseguenza. Comunica gli aggiornamenti procedurali a tutte le parti interessate e ai team. Converti le operazioni in gioco per condividere le best practice e fornire occasioni di formazione ai team.

- Prevedi gli insuccessi: massimizza il successo operativo definendo scenari di insuccesso per comprendere il profilo di rischio del carico di lavoro e il suo impatto sui risultati aziendali. Testa l'efficacia delle procedure e la risposta del team a questi errori simulati. Prendi decisioni informate per gestire i rischi aperti identificati tramite i test.
- Impara da tutti i parametri e gli eventi operativi: favorisci il miglioramento tramite le lezioni apprese da tutti gli eventi e gli errori operativi. Condividi ciò che hai imparato con i vari team e con tutta l'organizzazione. Gli insegnamenti evidenziano dati e aneddoti su come le operazioni contribuiscono al conseguimento dei risultati aziendali.
- Utilizza servizi gestiti: riduci il carico operativo utilizzando servizi gestiti AWS, laddove possibile. Sviluppa procedure operative basate sulle interazioni con tali servizi.

Definizione

Esistono quattro aree di best practice per l'eccellenza operativa nel cloud:

- Organizzazione
- Preparazione
- Gestione
- Evoluzione

La leadership dell'organizzazione definisce gli obiettivi aziendali. La tua organizzazione deve comprendere i requisiti e le priorità e utilizzarli per organizzare e condurre attività a supporto del raggiungimento dei risultati aziendali. Il carico di lavoro deve generare le informazioni necessarie per supportarlo. L'implementazione di servizi per ottenere l'integrazione, l'implementazione e la distribuzione del carico di lavoro, darà vita a un flusso maggiore di modifiche vantaggiose in fase di produzione attraverso l'automazione dei processi ripetitivi.

Potrebbero esserci rischi inerenti al funzionamento del carico di lavoro. Occorre comprendere questi rischi e prendere una decisione consapevole prima di passare alla fase di produzione. I team devono essere in grado di supportare il carico di lavoro. Le metriche aziendali e operative derivate dai risultati aziendali desiderati ti aiuteranno a comprendere lo stato del carico di lavoro e le attività operative e di rispondere agli incidenti. Le priorità cambieranno di pari passo con l'evoluzione delle esigenze

aziendali e dell'ambiente aziendale. Utilizza questi aspetti come ciclo di feedback per apportare continui miglioramenti all'organizzazione e alle operazioni legate al carico di lavoro.

Best practice

> **Note**
>
> Tutte le domande sull'eccellenza operativa hanno il prefisso OPS come abbreviazione del principio.

Argomenti

Organizzazione

È necessario che i team abbiano una comprensione condivisa dell'intero carico di lavoro, del loro ruolo rispetto al carico di lavoro, nonché degli obiettivi aziendali condivisi. In questo modo potranno stabilire le priorità che possono favorire il successo aziendale. Un'adeguata definizione delle priorità massimizzerà i risultati dei tuoi sforzi. Valuta le esigenze dei clienti interni ed esterni coinvolgendo le principali parti interessate, compresi i team aziendali, di sviluppo e operativi, per stabilire dove concentrare le attività operative. Valutando le esigenze dei clienti otterrai una conoscenza approfondita del supporto necessario per raggiungere i risultati aziendali. Accertati di essere a conoscenza delle linee guida o degli obblighi definiti dalla governance organizzativa e da fattori esterni, come i requisiti di conformità normativa e gli standard di settore, che possono imporre o accentuare un'attenzione specifica. Accertati di disporre di meccanismi per identificare le modifiche ai requisiti di governance interna e di conformità esterni. Se non viene identificato alcun requisito, conferma l'applicazione della due diligence per giungere a tale determinazione. Rivedi regolarmente le tue priorità in modo che possano essere aggiornate al mutare delle esigenze.

Valuta le minacce per l'azienda (ad esempio rischi e responsabilità aziendali e minacce alla sicurezza delle informazioni) e conserva queste informazioni in un registro dei rischi. Valuta l'impatto dei rischi e dei compromessi tra interessi concorrenti o approcci alternativi. Ad esempio, accelerare l'introduzione

sul mercato di nuove funzionalità può essere preferibile all'ottimizzazione dei costi. Oppure, è possibile scegliere un database relazionale per i dati non relazionali per semplificare l'iniziativa di migrazione di un sistema senza rifattorizzare. Gestisci i vantaggi e i rischi per prendere decisioni informate nel determinare dove concentrare gli sforzi. Alcuni rischi o scelte possono essere accettabili per un certo periodo di tempo, potrebbe essere possibile ridurre i rischi associati o la presenza di un rischio potrebbe diventare inaccettabile, nel qual caso si intraprenderà un'azione per risolverlo.

I tuoi team devono comprendere quale contributo offrono nel raggiungimento dei risultati aziendali. I team devono avere obiettivi condivisi e comprendere il proprio ruolo nel successo degli altri team. Comprendere la responsabilità, la proprietà, il modo in cui vengono prese le decisioni e chi ha l'autorità decisionale aiuterà a concentrare gli sforzi e a ottimizzare i contributi dei team. Le esigenze di un team sono influenzate dal cliente supportato, dall'organizzazione, dalla composizione del team e dalle caratteristiche del carico di lavoro. Non è ragionevole aspettarsi che un singolo modello operativo sia in grado di supportare tutti i team e i relativi carichi di lavoro dell'organizzazione.

Assicurati che siano identificati i responsabili di ogni applicazione, carico di lavoro, piattaforma e componente dell'infrastruttura e che per ogni processo e procedura sia identificato un responsabile della definizione e dei responsabili delle prestazioni.

La comprensione del valore aziendale di ogni componente, processo e procedura, del motivo per cui tali risorse sono presenti o le attività vengono eseguite e del perché tale proprietà esiste indirizzerà le azioni dei membri del team. Definisci chiaramente le responsabilità dei membri del team in modo che possano agire in modo appropriato e disporre di meccanismi per identificare responsabilità e proprietà. Implementa meccanismi per richiedere aggiunte, modifiche ed eccezioni in modo da non porre limiti all'innovazione. Definisci gli accordi tra i team che descrivono il modo in cui collaborano per supportarsi reciprocamente e contribuire ai risultati aziendali.

Fornisci supporto ai membri del team in modo che possano essere più efficaci nell'azione e nel supporto dei risultati aziendali. La leadership aziendale di alto livello deve stabilire le aspettative e misurare il successo. La leadership aziendale di alto livello è promotrice, sostenitrice e motore per l'adozione delle best practice e l'evoluzione dell'organizzazione. Consenti ai membri del team di intervenire quando i risultati sono a rischio per ridurre al minimo l'impatto e incoraggiali a rivolgersi ai responsabili decisionali e alle parti interessate quando ritengono che esista un rischio, in modo da poterlo risolvere e prevenire gli incidenti. Fornisci comunicazioni tempestive, chiare e concrete dei rischi noti e degli eventi pianificati in modo che i membri del team possano agire in modo tempestivo e appropriato.

Incoraggia la sperimentazione per accelerare l'apprendimento e mantenere i membri del team interessati e coinvolti. I team devono aumentare le proprie competenze per adottare nuove

tecnologie e supportare i cambiamenti della domanda e delle responsabilità. Fornisci il tuo supporto e incoraggiamento offrendo tempo strutturato dedicato per l'apprendimento. Assicurati che i membri del team dispongano delle risorse, in termini sia di strumenti sia di membri del team, per avere successo e adattarsi, sostenendo i risultati aziendali. Sfrutta la diversità tra organizzazioni per cercare più prospettive uniche. Usa questa prospettiva per incrementare l'innovazione, mettere in discussione le tue ipotesi e ridurre il rischio di bias confermativi. Aumenta l'inclusione, la diversità e l'accessibilità all'interno dei team per ottenere prospettive vantaggiose.

Se esistono requisiti normativi e di conformità esterni applicabili alla tua organizzazione, utilizza le risorse fornite da AWS Cloud Compliance per promuovere la formazione dei tuoi team affinché siano in grado di valutare il relativo impatto sulle tue priorità. Il Framework Well-Architected enfatizza formazione, misurazione e miglioramento Fornisce una strategia coerente per la valutazione delle architetture e l'implementazione di progetti in grado di dimensionarsi nel corso del tempo. AWS mette a disposizione AWS Well-Architected Tool per aiutarti ad analizzare il tuo approccio prima dello sviluppo e lo stato dei tuoi carichi di lavoro prima e durante la fase di produzione. Puoi confrontare i carichi di lavoro con le best practice architetturali AWS più recenti, monitorarne lo stato generale e ottenere informazioni sui potenziali rischi. AWS Trusted Advisor è uno strumento che fornisce l'accesso a una serie di controlli di base che propongono ottimizzazioni utili per la definizione delle tue priorità. I clienti del supporto Business ed Enterprise hanno accesso a ulteriori controlli a livello di sicurezza, affidabilità, prestazioni, ottimizzazione dei costi e sostenibilità che possono essere utili per definire le loro priorità.

AWS può aiutarti a sensibilizzare i team su AWS e i suoi servizi, affinché comprendano meglio l'impatto delle loro scelte sul carico di lavoro. Per istruire i tuoi team, utilizza le risorse fornite da Supporto AWS (Centro conoscenze AWS, AWS Discussion Forums e Supporto AWS Center) e la documentazione AWS. Contatta Supporto AWS attraverso il Supporto AWS Center per assistenza sulle tue domande su AWS. AWS condivide inoltre le best practice e i modelli appresi attraverso la gestione di AWS nella Amazon Builders' Library. Un'ampia gamma di ulteriori informazioni utili è disponibile tramite il blog AWS e il podcast ufficiale di AWS. AWS Training and Certification offre risorse di formazione tramite corsi digitali gestiti dall'utente sulle nozioni di base di AWS. Per supportare ulteriormente lo sviluppo delle competenze AWS del tuo team, è anche possibile iscriversi a corsi di formazione con istruttore.

Per facilitare la gestione dei modelli operativi, è consigliabile utilizzare strumenti o servizi che consentano di gestire centralmente gli ambienti su più account, ad esempio AWS Organizations. Servizi come AWS Control Tower ampliano questa funzionalità di gestione consentendoti di definire blueprint (a supporto dei tuoi modelli operativi) per configurare gli account, applicare la governance continua tramite AWS Organizations e automatizzare il provisioning di nuovi account. I fornitori

di servizi gestiti, come AWS Managed Services, partner AWS Managed Services o i fornitori di servizi gestiti della rete dei partner AWS offrono esperienza nell'implementazione di ambienti cloud e supportano i requisiti di sicurezza e conformità e gli obiettivi aziendali. L'aggiunta di servizi gestiti al tuo modello operativo ti consente di risparmiare tempo e risorse e ti permette di mantenere i team interni snelli e focalizzati sui risultati strategici che differenzieranno la tua attività, anziché sullo sviluppo di nuove competenze e funzionalità.

Le seguenti domande si concentrano su queste considerazioni relative all'eccellenza operativa. Per l'elenco completo delle domande e delle best practice relative all'eccellenza operativa, consulta l'Appendice.

OPS 1: in che modo stabilisci quali sono le tue priorità?

È necessario che ognuno comprenda il proprio ruolo nel conseguimento del successo aziendale . Devi disporre di obiettivi comuni al fine di stabilire le priorità per le risorse. Ciò massimizzerà i risultati dei tuoi sforzi.

OPS 2: in che modo strutturi la tua organizzazione per supportare i risultati aziendali?

I tuoi team devono comprendere quale contributo offrono nel raggiungimento dei risultati aziendali. I team devono avere obiettivi condivisi e comprendere il proprio ruolo nel successo degli altri team. Comprendere la responsabilità, la proprietà, il modo in cui vengono prese le decisioni e chi ha l'autorità decisionale aiuterà a concentrare gli sforzi e a ottimizzare i contributi dei team.

OPS 3: in che modo la cultura aziendale supporta i risultati aziendali?

Fornisci supporto ai membri del team in modo che possano essere più efficaci nell'azione e nel supporto dei risultati aziendali.

Ad esempio, a un certo punto potresti realizzare che desideri dare maggiore risalto a un piccolo sottoinsieme delle tue priorità. Utilizza un approccio equilibrato nel lungo termine per garantire lo sviluppo delle capacità necessarie e la gestione del rischio. Rivedi regolarmente le tue priorità e aggiornale al mutare delle esigenze. Quando la responsabilità e la proprietà sono indefinite o sconosciute, rischi sia di non affrontare tempestivamente le attività necessarie sia di adoperarti in

modo ridondante e potenzialmente conflittuale per rispondere a tali esigenze. La cultura organizzativa influisce direttamente sulla soddisfazione sul lavoro e sulla conservazione dei membri del team. Sostieni il coinvolgimento e le capacità dei membri del tuo team per ottenere il successo della tua attività. La sperimentazione è necessaria per realizzare l'innovazione e trasformare le idee in risultati. Un risultato indesiderato è un esperimento riuscito che ha identificato un percorso che non porterà al successo.

Preparazione

Per prepararti all'eccellenza operativa devi comprendere i carichi di lavoro e i loro comportamenti previsti. Sarai dunque in grado di progettare i carichi di lavoro in modo tale che forniscano informazioni sul loro stato e di creare le procedure per supportarli adeguatamente.

Progetta il tuo carico di lavoro affinché ti fornisca le informazioni necessarie a comprenderne lo stato interno (ad esempio, parametri, log, eventi e tracce) in tutti i componenti a supporto dell'osservabilità e dell'analisi dei problemi. L'osservabilità va oltre il semplice monitoraggio, in quanto fornisce una comprensione completa del funzionamento interno di un sistema basata sui suoi output esterni. L'osservabilità è legata a doppio filo a metriche, log e tracce per offrire informazioni approfondite sul comportamento e sulle dinamiche del sistema. Grazie a un'osservabilità efficace, i team possono distinguere modelli, anomalie e tendenze, così da essere in grado di affrontare in modo proattivo potenziali problemi e mantenere l'integrità del sistema. L'identificazione degli indicatori chiave di prestazione (KPI) è fondamentale per garantire l'allineamento tra le attività di monitoraggio e gli obiettivi aziendali. Questo allineamento garantisce che i team prendano decisioni basate sui dati e su metriche realmente importanti, ottimizzando sia le prestazioni del sistema sia i risultati aziendali. Inoltre, l'osservabilità consente alle aziende di essere proattive anziché reattive. I team possono comprendere le relazioni causa-effetto all'interno dei loro sistemi, prevedendo e prevenendo i problemi anziché limitarsi a reagire quando si verificano. Con l'evolversi dei carichi di lavoro, è essenziale riesaminare e perfezionare la strategia di osservabilità, assicurandosi che rimanga pertinente ed efficace.

Adotta strategie che migliorino il flusso delle modifiche in produzione e che consentano la rifattorizzazione, il feedback veloce sulla qualità e la correzione di errori. Tali prassi accelerano l'ingresso in produzione delle modifiche vantaggiose, limitano i problemi distribuiti e consentono una rapida identificazione e risoluzione dei problemi introdotti attraverso le attività di implementazione o scoperti negli ambienti.

Adotta prassi per fornire un feedback rapido sulla qualità e che permettano un ripristino veloce dalle modifiche che non hanno i risultati previsti. L'uso di queste prassi consente di mitigare l'impatto dei

problemi introdotti attraverso l'implementazione delle modifiche. Prepara un piano in caso di esito negativo delle modifiche in modo da poter rispondere più rapidamente se necessario, testando e convalidando le modifiche apportate. Sii consapevole delle attività pianificate nei tuoi ambienti in modo da poter gestire il rischio di modifiche che influiscono sulle attività pianificate. Privilegia le modifiche frequenti, piccole e reversibili per limitarne l'ambito. In questo modo velocizzerai risoluzione dei problemi e correzione, mantenendo la possibilità di rollback delle modifiche. In tal modo, è anche possibile ottenere più frequentemente i vantaggi offerti dalle modifiche importanti.

Valuta la prontezza operativa del carico di lavoro, dei processi e delle procedure, nonché del personale, per comprendere i rischi operativi correlati al carico di lavoro. Utilizza un processo omogeneo (inclusi elenchi di controllo manuali o automatici) per sapere quando puoi rilasciare un carico di lavoro o una modifica. Questo inoltre ti aiuterà a trovare le eventuali aree che necessitano di pianificazioni. Predisponi runbook che documentino le tue attività di routine e manuali alla base dei processi per la risoluzione dei problemi. Analizza i vantaggi e i rischi per prendere decisioni informate e consentire l'adozione delle modifiche nella produzione.

In AWS, puoi vedere il tuo carico di lavoro completo (applicazioni, infrastruttura, policy, governance e operazioni) in forma di codice. In tal modo è possibile applicare la stessa disciplina ingegneristica utilizzata per il codice dell'applicazione a ogni elemento dello stack, condividendoli tra team o organizzazioni per sfruttare al massimo i vantaggi delle attività di sviluppo. Utilizza le operazioni come codice nel cloud e sfrutta la possibilità di sperimentare per sviluppare il tuo carico di lavoro e le procedure operative ed esercitarti con gli errori in modo sicuro. CloudFormation ti consente di avere ambienti di sviluppo, di test e di produzione sandbox, omogenei e basati su modelli, con livelli crescenti di controllo operativo.

Le seguenti domande si concentrano su queste considerazioni relative all'eccellenza operativa.

OPS 4: in che modo implementi l'osservabilità nel carico di lavoro?

Implementare l'osservabilità nel carico di lavoro ti permette di comprendere lo stato di quest'ultimo e di adottare decisioni basate sui dati e che riflettono i requisiti aziendali.

OPS 5: in che modo riduci i difetti, favorisci la correzione e migliori il flusso nella produzione?

Adotta strategie che migliorino il flusso delle modifiche in produzione e che favoriscano la rifattori zzazione, il feedback veloce sulla qualità e la correzione di errori. Tali approcci accelerano l'ingresso in produzione delle modifiche vantaggiose, contengono i problemi che si sono diffusi e

OPS 5: in che modo riduci i difetti, favorisci la correzione e migliori il flusso nella produzione?

permettono di ottenere una rapida identificazione e risoluzione dei problemi introdotti attraverso le attività di implementazione.

OPS 6: in che modo mitighi i rischi dell'implementazione?

Adotta approcci per fornire un feedback rapido sulla qualità e che permettano un ripristino veloce dalle modifiche che non hanno i risultati previsti. L'uso di queste prassi consente di mitigare l'impatto dei problemi introdotti attraverso l'implementazione delle modifiche.

OPS 7: come fai a sapere che sei pronto a supportare un carico di lavoro?

Valuta la prontezza operativa del carico di lavoro, dei processi e delle procedure, nonché del personale per comprendere i rischi operativi correlati al carico di lavoro.

Investi nell'implementazione di attività operative come codice per aumentare al massimo la produttività del personale operativo, ridurre al minimo la frequenza degli errori e consentire risposte automatizzate. Utilizza l'analisi prefallimentare per prevedere errori e creare procedure ove opportuno. Applica i metadati utilizzando i tag delle risorse e i AWS Resource Groups seguendo una strategia di applicazione dei tag coerente per consentire l'identificazione delle risorse. Applica tag alle risorse per organizzare, monitorare i costi e controllare gli accessi e ottimizza l'esecuzione delle attività operative automatizzate. Adotta procedure di distribuzione che sfruttino l'elasticità del cloud per facilitare le attività di sviluppo e la pre-distribuzione dei sistemi e avere implementazioni più rapide. Quando apporti modifiche agli elenchi di controllo che utilizzi per valutare i tuoi carichi di lavoro, pianifica quello che farai con i sistemi live che non risultano più conformi.

Gestione

L'osservabilità ti consente di concentrarti su dati significativi e di comprendere le interazioni e l'output del tuo carico di lavoro. Concentrandoti sugli approfondimenti essenziali ed eliminando i dati non necessari, mantieni un approccio diretto alla comprensione delle prestazioni del carico di lavoro. È essenziale non solo raccogliere dati, ma anche interpretarli correttamente. Definisci linee guida chiare, imposta soglie di avviso appropriate e monitora attivamente eventuali deviazioni. Un cambiamento in una metrica chiave, specialmente se correlata ad altri dati, permette di individuare

aree problematiche specifiche. Grazie all'osservabilità hai strumenti per prevedere e affrontare potenziali sfide, assicurando che il tuo carico di lavoro funzioni senza intoppi e soddisfi le esigenze aziendali.

La corretta operatività di un carico di lavoro è misurata dal raggiungimento di risultati per l'azienda e per i clienti. Definisci i risultati desiderati, determina in che modo verrà misurato il successo e individua i parametri che saranno usati nei calcoli per determinare se il carico di lavoro e le operazioni sono efficaci. L'integrità delle operazioni include sia lo stato del carico di lavoro sia lo stato e il successo delle operazioni a supporto del carico di lavoro (ad esempio, l'implementazione e la risposta agli incidenti). Stabilisci le basi dei parametri per migliorare, eseguire indagini e intervenire, raccogliere e analizzare i parametri, quindi conferma la tua comprensione del successo operativo e della sua evoluzione nel corso del tempo. Usa i parametri raccolti per determinare il grado di soddisfazione dei clienti, capire se stai rispondendo alle esigenze aziendali e individuare gli aspetti da migliorare.

La gestione efficiente ed efficace degli eventi operativi è fondamentale per raggiungere l'eccellenza operativa. Ciò si applica agli eventi operativi sia pianificati che non. Usa runbook precisi per gli eventi chiari e ricorri ai playbook per favorire l'analisi e la risoluzione degli altri eventi. Attribuisci la priorità alle risposte agli eventi in base al loro impatto sull'azienda e sui clienti. Assicurati che, in caso di avvisi in risposta a un evento, vi sia una procedura associata da seguire, con un proprietario ben preciso. Definisci in anticipo il personale richiesto per risolvere un evento e includi dei processi di escalation per coinvolgere altro personale, ove necessario, in base all'urgenza e all'impatto. Individua e coinvolgi le persone che hanno l'autorità per prendere decisioni in merito alle linee d'azione laddove vi sia un impatto aziendale dovuto a una risposta a un evento non gestito precedentemente.

Comunica lo stato operativo dei carichi di lavoro tramite pannelli di controllo e notifiche personalizzati in base al pubblico di destinazione (ad esempio cliente, azienda, sviluppatori, addetti alle operazioni), in modo che gli interessati possano agire in maniera adeguata, che le loro aspettative vengano soddisfatte e che siano informati sulla ripresa delle normali operazioni.

In AWS puoi generare panoramiche di pannelli di controllo per i parametri raccolti dai carichi di lavoro e in modo nativo da AWS. Puoi sfruttare CloudWatch o applicazioni di terze parti per aggregare e presentare panoramiche a livello di business, di carico di lavoro e di operazioni delle attività operative. AWS fornisce approfondimenti sui carichi di lavoro attraverso funzionalità di creazione di log, tra cui AWS X-Ray, CloudWatch, CloudTrail e log di flusso VPC, che consentono di identificare i problemi del carico di lavoro a supporto dell'analisi delle cause principali e della risoluzione dei problemi.

Le seguenti domande si concentrano su queste considerazioni relative all'eccellenza operativa.

OPS 8: in che modo utilizzi l'osservabilità del carico di lavoro nella tua organizzazione?

Garantire l'integrità del carico di lavoro sfruttando l'osservabilità. Utilizzare metriche, log e tracce pertinenti per ottenere una visione completa delle prestazioni del carico di lavoro e risolvere i problemi in modo efficiente.

OPS 9: come fai a comprendere lo stato delle operazioni?

Definisci, acquisisci e analizza i parametri delle operazioni per ottenere visibilità sugli eventi delle operazioni, in modo da intraprendere le azioni appropriate.

OPS 10: in che modo gestisci gli eventi del carico di lavoro e delle operazioni?

Prepara e convalida le procedure in risposta agli eventi per ridurre al minimo il loro impatto sul tuo carico di lavoro.

Tutti i parametri raccolti devono essere allineati alle esigenze aziendali e ai risultati che supportano. Sviluppa risposte con script per eventi ben compresi e automatizza le prestazioni in risposta al riconoscimento dell'evento.

Evoluzione

Impara, condividi e migliora continuamente per sostenere l'eccellenza operativa. Dedica dei cicli di lavoro al raggiungimento di miglioramenti incrementali quasi continui. Esegui l'analisi post-incidente di tutti gli eventi che influiscono sul cliente. Identifica i fattori che contribuiscono e le azioni preventive per limitare o prevenire la ricorrenza. Comunica i fattori che contribuiscono alle comunità interessate, nel modo più adeguato. Valuta regolarmente e assegna le priorità alle opportunità di miglioramento (ad esempio, richieste di funzionalità, risoluzione dei problemi e requisiti di conformità), includendo sia il carico di lavoro sia le procedure operative.

Includi i loop di feedback nelle tue procedure per individuare rapidamente gli aspetti che devono essere migliorati e per acquisire conoscenze dall'esecuzione delle operazioni.

Condividi le lezioni apprese con i vari team per condividerne anche i vantaggi. Analizza le tendenze all'interno delle lezioni apprese ed esegui analisi trasversali retrospettive dei parametri operativi per

individuare le opportunità e i metodi di miglioramento. Implementa le modifiche previste per garantire il miglioramento e valuta i risultati per favorire il successo.

In AWS, è possibile esportare i dati di log in Amazon S3 o inviare log direttamente ad Amazon S3 per lo storage a lungo termine. Con AWS Glue puoi individuare e preparare i dati di log in Amazon S3 per l'analisi e archiviare i metadati associati nel AWS Glue Data Catalog. Grazie all'integrazione nativa con AWS Glue, quindi, Amazon Athena può essere utilizzato per analizzare i dati di log, eseguendo query tramite SQL standard. Con uno strumento di business intelligence come Amazon Quick puoi visualizzare, esplorare e analizzare i tuoi dati. Rilevamento di tendenze ed eventi di interesse che possono portare a miglioramenti.

La seguente domanda si concentra su queste considerazioni relative all'eccellenza operativa.

OPS 11: in che modo fai evolvere le operazioni?

Dedica tempo e risorse per ottenere un miglioramento incrementale pressoché continuo, per far evolvere l'efficacia e l'efficienza delle tue operazioni.

L'evoluzione efficace delle operazioni si basa sugli elementi seguenti: miglioramenti piccoli ma frequenti; creazione di ambienti sicuri e tempo per sperimentare, sviluppare e testare i miglioramenti; ambienti in cui le persone siano incoraggiate a imparare dagli errori. Il supporto alle operazioni per ambienti sandbox, di sviluppo, di prova e di produzione, con un crescente livello di controlli operativi, facilita lo sviluppo e aumenta la prevedibilità dei risultati positivi dalle modifiche passate in produzione.

Risorse

Consulta le seguenti risorse per ulteriori informazioni sulle best practice relative all'eccellenza operativa.

Documentazione

- DevOps e AWS

Whitepaper

- Pilastro dell'eccellenza operativa

Video

- DevOps in Amazon

Sicurezza

Il pilastro della sicurezza contempla la capacità di proteggere dati, sistemi e asset per sfruttare le tecnologie cloud in modo da migliorare la sicurezza.

Il pilastro della sicurezza offre una panoramica dei principi di progettazione, delle best practice e delle domande. Puoi trovare linee guida prescrittive sull'implementazione nel whitepaper sul pilastro della sicurezza.

Argomenti

- Principi di progettazione
- Definizione
- Best practice
- Risorse

Principi di progettazione

Nel cloud sono presenti diversi principi utili per rafforzare la sicurezza del carico di lavoro:

- Implementazione di una solida base di identità: implementa il principio del privilegio minimo e applica la separazione dei compiti assegnando l'autorizzazione appropriata per ogni interazione con le risorse AWS. Centralizza la gestione delle identità e mira a eliminare la dipendenza dalle credenziali statiche a lungo termine.
- Mantenimento della tracciabilità: monitora, crea avvisi e verifica in tempo reale le operazioni e le modifiche apportate al tuo ambiente. Integra la raccolta di log e parametri con i sistemi per analizzare e intervenire automaticamente.
- Applicazione della sicurezza a tutti i livelli: applica un approccio di difesa avanzata con più controlli di sicurezza. Applicalo a tutti i livelli (ad esempio, edge di rete, VPC, bilanciamento del carico, ogni istanza e servizio di elaborazione, sistema operativo, applicazione e codice).
- Automatizzazione delle best practice di sicurezza: i meccanismi di sicurezza automatizzati basati su software migliorano la capacità di scalare le risorse in modo sicuro, più rapido e conveniente.

Crea architetture sicure, compresa l'implementazione dei controlli, definite e gestite come codice nei modelli controllati dalle versioni.

- Protezione dei dati in transito e a riposo: classifica i dati in base a livelli di sensibilità e utilizza meccanismi quali crittografia, tokenizzazione e controllo degli accessi, ove opportuno.
- Accesso limitato delle persone ai dati: utilizza meccanismi e strumenti per ridurre o eliminare l'esigenza di accesso diretto o di elaborazione manuale dei dati. Ciò riduce il rischio di perdita, modifica e altri errori umani durante la gestione dei dati sensibili.
- Preparazione agli eventi di sicurezza: preparati per un incidente creando policy e processi di analisi e gestione degli incidenti in linea con i requisiti dell'organizzazione. Esegui simulazioni di risposta agli incidenti e utilizza strumenti dotati di automazione per aumentare la velocità nel rilevamento, nell'indagine e nel ripristino.

Definizione

Esistono sette aree di best practice per la sicurezza nel cloud.

- Nozioni di base sulla sicurezza
- Gestione dell'identità e degli accessi
- Rilevamento
- Protezione dell'infrastruttura
- Protezione dei dati
- Risposta agli incidenti
- Sicurezza delle applicazioni

Prima di progettare qualsiasi carico di lavoro, è necessario implementare pratiche che influenzano la sicurezza. Dovrai controllare chi può fare cosa. Inoltre, devi essere in grado di identificare gli incidenti di sicurezza, proteggere i tuoi sistemi e i tuoi servizi e mantenere la riservatezza e l'integrità dei dati attraverso la loro protezione. Dovresti avere dei processi ben definiti e rodati per rispondere a eventuali problemi di sicurezza. Questi strumenti e tecniche sono importanti perché supportano obiettivi come la prevenzione delle perdite finanziarie o la conformità agli obblighi normativi.

Il modello di responsabilità condivisa di AWS aiuta le organizzazioni che adottano il cloud a raggiungere i loro obiettivi in termini di sicurezza e conformità. Dato che AWS mette fisicamente in sicurezza l'infrastruttura che supporta i nostri servizi cloud, come cliente AWS puoi concentrarti

sull'utilizzo dei servizi per raggiungere gli obiettivi. Il cloud AWS fornisce, inoltre, l'accesso ai dati sulla sicurezza e offre un approccio automatico per rispondere agli eventi di sicurezza.

Best practice

Argomenti

Nozioni di base sulla sicurezza

La seguente domanda si concentra su queste considerazioni relative alla sicurezza. Per l'elenco completo delle domande e delle best practice relative alla sicurezza, consulta l'Appendice.

SEC 1: come gestisci in modo sicuro un carico di lavoro?
Per gestire il carico di lavoro in modo sicuro, è necessario applicare le best practice globali a ogni area di sicurezza. Segui i requisiti e i processi definiti in termini di eccellenza operativa a livello organizzativo e di carico di lavoro e applicali a tutte le aree. Rimanere aggiornati con le raccomandazioni di AWS, le fonti di settore nonché con l'intelligence sulle minacce aiuta a sviluppare il modello di rischio e gli obiettivi di controllo. L'automazione dei processi di sicurezza, i test e la convalida permettono di dimensionare le operazioni di sicurezza.

In AWS, è consigliabile la segregazione dei vari carichi di lavoro per account, in base alla loro funzione e ai requisiti di conformità o di sensibilità dei dati.

Gestione dell'identità e degli accessi

La gestione delle identità e degli accessi è una parte principale di un programma di sicurezza delle informazioni e garantisce che solo gli utenti e i componenti autorizzati e autenticati possano accedere

alle tue risorse e solo nella modalità che hai stabilito. Ad esempio, è necessario definire i principali (ovvero account, utenti, ruoli e servizi che possono eseguire operazioni nel tuo account), creare policy allineate a tali principali e implementare una forte gestione delle credenziali. Questi elementi a gestione privilegiata formano i concetti chiave dell'autenticazione e dell'autorizzazione.

In AWS, la gestione dei privilegi è principalmente supportata dal servizio AWS Identity and Access Management (IAM), che consente di controllare l'accesso utente e l'accesso programmatico ai servizi e alle risorse AWS. È necessario applicare criteri granulari che assegnano autorizzazioni a un utente, gruppo, ruolo o risorsa. Hai anche la possibilità di richiedere pratiche di password complesse, come il livello di complessità, evitare il riutilizzo e applicare l'autenticazione a più fattori (MFA). È possibile utilizzare la federazione con il servizio di directory esistente. Per i carichi di lavoro che richiedono che i sistemi abbiano accesso ad AWS, IAM consente l'accesso sicuro tramite ruoli, profili dell'istanza, federazione delle identità e credenziali temporanee.

Le seguenti domande si concentrano su queste considerazioni relative alla sicurezza.

SEC 2: come gestisci l'autenticazione per persone e macchine?

Ci sono due tipi di identità da gestire quando inizi a utilizzare carichi di lavoro AWS sicuri. Comprendere il tipo di identità necessaria per gestire e concedere l'accesso ti aiuta a verificare che le identità corrette abbiano accesso alle risorse giuste nelle condizioni adeguate.

Identità umane: amministratori, sviluppatori, operatori e utenti finali necessitano di un'identità per accedere agli ambienti e alle applicazioni AWS. Si tratta di membri dell'organizzazione o di utenti esterni con cui collabori e che interagiscono con le risorse AWS tramite Web browser, applicazioni client o strumenti a riga di comando interattivi.

Identità di macchine: le applicazioni di servizio, gli strumenti operativi e i carichi di lavoro necessita no di un'identità per effettuare richieste ai servizi AWS, ad esempio per leggere i dati. Queste identità includono macchine in esecuzione nei tuoi ambienti AWS, ad esempio le istanze Amazon EC2 o le funzioni AWS Lambda. Puoi gestire le identità di macchine anche per soggetti esterni che necessitano dell'accesso. Inoltre, potresti disporre di macchine al di fuori di AWS che devono accedere al tuo ambiente AWS.

SEC 3: come gestisci le autorizzazioni per persone e macchine?

Gestisci le autorizzazioni per controllare l'accesso alle identità di persone e macchine che richiedono l'accesso ad AWS e al tuo carico di lavoro. Le autorizzazioni controllano chi può accedere a cosa e a quali condizioni.

Le credenziali non devono essere condivise tra nessun utente o sistema. L'accesso degli utenti dovrebbe essere concesso utilizzando un approccio del privilegio minimo con le best practice, inclusi i requisiti di password e l'applicazione del MFA. L'accesso programmatico, comprese le chiamate API ai servizi AWS, deve essere eseguito utilizzando credenziali temporanee e con privilegi limitati come quelle emesse da AWS Security Token Service.

Gli utenti hanno bisogno di un accesso programmatico se desiderano interagire con AWS esternamente a Console di gestione AWS. La modalità con cui concedere l'accesso programmatico dipende dal tipo di utente che accede ad AWS.

Per fornire agli utenti l'accesso programmatico, scegli una delle seguenti opzioni.

Quale utente necessita dell'accesso programmatico?	Per	Come
IAM	(Consigliato) Utilizza credenziali della console come credenziali temporanee per firmare richieste programmatiche alla AWS CLI, agli SDK AWS o alle API AWS.	Segui le istruzioni per l'interfaccia che desideri utilizzare. • Per AWS CLI, consulta Accesso allo sviluppo locale di AWS nella Guida per l'utente di AWS Command Line Interface. • Per gli SDK AWS, consulta Accesso per lo sviluppo AWS locale nella Guida di riferimento agli SDK e agli strumenti AWS.
Identità della forza lavoro	Utilizza credenziali temporanee per firmare richieste	Segui le istruzioni per l'interfaccia che desideri utilizzare.

Quale utente necessita dell'accesso programmatico?	Per	Come
(Utenti gestiti nel centro identità IAM)	programmatiche alla AWS CLI, agli SDK AWS o alle API AWS.	• Per la AWS CLI, consulta la pagina Configurazione della AWS CLI per l'uso di AWS IAM Identity Center nella Guida per l'utente dell'AWS Command Line Interface. • Per gli SDK AWS, gli strumenti e le API AWS, consulta la pagina Autenticazione Centro identità IAM nella Guida di riferimento per SDK e strumenti AWS.
IAM	Utilizza credenziali temporanee per firmare richieste programmatiche alla AWS CLI, agli SDK AWS o alle API AWS.	Segui le istruzioni in Utilizzo di credenziali temporanee con le risorse AWS nella Guida per l'utente IAM.

Quale utente necessita dell'accesso programmatico?	Per	Come
IAM	(Non consigliato) Utilizza credenziali a lungo termine per firmare richieste programmatiche alla AWS CLI, agli SDK AWS o alle API AWS.	Segui le istruzioni per l'interfaccia che desideri utilizzare. • Per la AWS CLI, consulta la pagina Autenticazione tramite credenziali utente IAM nella Guida per l'utente dell'AWS Command Line Interface. • Per gli SDK e gli strumenti AWS, consulta la pagina Autenticazione con credenziali a lungo termine nella Guida di riferimento per SDK e strumenti AWS. • Per le API AWS, consulta la pagina Gestione delle chiavi di accesso per utenti IAM nella Guida per l'utente IAM.

AWS offre risorse utili per la gestione di identità e accessi. Per imparare le best practice, scopri i nostri laboratori pratici sulla gestione delle credenziali e dell'autenticazione, sul controllo dell'accesso umano e sul controllo dell'accesso programmatico.

Rilevamento

Puoi utilizzare i controlli di rilevamento per identificare una potenziale minaccia o un potenziale incidente di sicurezza. Questi controlli sono una parte essenziale dei framework di governance e possono essere utilizzati per supportare il processo di qualità o un obbligo legale o di conformità e per l'identificazione delle minacce e gli sforzi nelle risposte. Ci sono diversi tipi di controlli di rilevamento. Ad esempio, la realizzazione di un inventario di risorse e dei loro attributi dettagliati promuove le decisioni più efficienti (e i controlli del ciclo di vita) per stabilire delle baseline operative. Puoi anche utilizzare audit interni, un esame dei controlli relativi ai sistemi di informazioni, per

verificare che le pratiche rispettino le policy e i requisiti e che tu abbia un set corretto di notifiche di avviso automatiche basate sulle condizioni definite. Questi controlli sono fattori di reazione importanti che possono aiutare la tua organizzazione a identificare e capire la portata dell'attività anomala.

In AWS, puoi implementare controlli investigativi elaborando log, eventi e monitoraggio che consentono audit, analisi automatizzate e notifiche. I log CloudTrail, le chiamate API AWS e CloudWatch forniscono il monitoraggio di parametri con notifiche, mentre AWS Config fornisce la cronologia delle configurazioni. Amazon GuardDuty è un servizio di rilevazione delle minacce che monitora costantemente possibili comportamenti dannosi o non autorizzati, così da proteggere i tuoi account e i tuoi carichi di lavoro su AWS. Sono inoltre disponibili log a livello di servizio, ad esempio puoi utilizzare Amazon Simple Storage Service (Amazon S3) per registrare le richieste di accesso.

La seguente domanda si concentra su queste considerazioni relative alla sicurezza.

SEC 4: in che modo individui ed esamini gli eventi di sicurezza?

Acquisisci e analizza gli eventi a partire da log e metriche per acquistare visibilità. Agisci su eventi di sicurezza e potenziali minacce per contribuire a rendere sicuro il carico di lavoro.

La gestione dei log è una parte importante di un carico di lavoro Well-Architected per ragioni che vanno da requisiti di sicurezza o forensi a disposizioni normative o legali. È fondamentale analizzare i log e rispondere in modo da identificare potenziali incidenti di sicurezza. AWS offre funzionalità che semplificano l'implementazione della gestione dei log, offrendo la possibilità di definire un ciclo di vita di conservazione dei dati o di definire dove verranno conservati, archiviati o eventualmente eliminati. Ciò rende la gestione dei dati prevedibile e affidabile, più semplice ed economica.

Protezione dell'infrastruttura

La protezione dell'infrastruttura comprende delle metodologie di controllo, come la difesa approfondita, necessarie per rispettare le best practice e gli obblighi organizzativi e normativi. L'utilizzo di queste metodologie è fondamentale per ottenere operazioni continuative e di successo sia nel cloud che on-premises.

In AWS, è possibile implementare l'ispezione di pacchetti stateful e stateless, sia utilizzando tecnologie native di AWS, sia utilizzando prodotti e servizi dei partner disponibili attraverso Marketplace AWS. È necessario utilizzare Amazon Virtual Private Cloud (Amazon VPC) per creare un ambiente privato, protetto e scalabile in cui è possibile definire la propria topologia, inclusi gateway, tabelle di routing e sottoreti pubbliche e private.

Le seguenti domande si concentrano su queste considerazioni relative alla sicurezza.

SEC 5: in che modo proteggi le risorse di rete?
Qualsiasi carico di lavoro che abbia una qualche forma di connettività di rete, che si tratti di Internet o di una rete privata, richiede più livelli di difesa per proteggere da minacce esterne e interne basate sulla rete.

SEC 6: in che modo proteggi le risorse di calcolo?
Le risorse di calcolo nel carico di lavoro richiedono più livelli di difesa per contribuire alla protezion e da minacce esterne e interne. Le risorse di calcolo includono istanze EC2, container, funzioni di AWS Lambda, servizi di database, dispositivi IoT e altro.

Si consigliano più livelli di difesa in qualsiasi tipo di ambiente. Nel caso della protezione dell'infrastruttura, molti concetti e metodi sono validi sia per modelli cloud che on-premises. L'applicazione della protezione dei confini, il monitoraggio dei punti di ingresso e di uscita e la creazione di log, il monitoraggio e le notifiche completi sono tutti elementi essenziali per un efficace piano di sicurezza delle informazioni.

I clienti AWS sono in grado di adattare o rafforzare la configurazione di Amazon Elastic Compute Cloud (Amazon EC2), di un container di Amazon Elastic Container Service (Amazon ECS) o di un'istanza AWS Elastic Beanstalk e mantenere questa configurazione su un'Amazon Machine Image (AMI) immutabile. Quindi, che siano avviati da Auto Scaling o lanciati manualmente, tutti i nuovi server virtuali (istanze) lanciati con questa AMI utilizzeranno la configurazione rafforzata.

Protezione dei dati

Prima di progettare qualsiasi sistema, devono essere stabiliti i requisiti fondamentali che influenzano la sicurezza. Ad esempio, la classificazione dei dati fornisce un modo per categorizzare i dati organizzativi basati sui livelli di sensibilità, mentre la crittografia protegge i dati evitandone l'intelligibilità per gli accessi non autorizzati. Questi strumenti e tecniche sono importanti perché supportano obiettivi come la prevenzione delle perdite finanziarie o la conformità agli obblighi normativi.

In AWS, le seguenti pratiche agevolano la protezione dei dati:

- Come cliente AWS mantieni il pieno controllo sui tuoi dati.
- AWS semplifica la crittografia dei dati e la gestione delle chiavi, inclusa la rotazione regolare delle chiavi, che può essere facilmente automatizzata da AWS o gestita da te.
- È disponibile la creazione di log dettagliati con contenuti importanti, come l'accesso ai file e le modifiche.
- AWS ha progettato sistemi di storage con una resilienza eccezionale. Ad esempio, Amazon S3 Standard, S3 Standard-IA, One Zone-IA S3 e Amazon Glacier sono tutti progettati per offrire una resistenza degli oggetti del 99,999999999% in un determinato anno. Questo livello di durabilità corrisponde a una perdita media annua prevista dello 0,000000001% di oggetti.
- Il controllo delle versioni, che può far parte di un più ampio processo di gestione del ciclo di vita dei dati, può proteggere da sovrascritture accidentali, eliminazioni e danni simili.
- AWS non avvia mai il trasferimento di dati tra regioni. Il contenuto inserito in una regione rimarrà in quella regione a meno che tu non utilizzi esplicitamente una funzionalità o sfrutti un servizio che fornisce tale funzionalità.

Le seguenti domande si concentrano su queste considerazioni relative alla sicurezza.

SEC 7: in che modo classifichi i dati?

La classificazione fornisce un modo per categorizzare i dati in base ai livelli di criticità e sensibilità, in modo da aiutarti a determinare i controlli di protezione e conservazione appropriati.

SEC 8: in che modo proteggi i dati a riposo?

Proteggi i dati a riposo implementando più controlli per ridurre il rischio di accessi non autorizzati o altri comportamenti impropri.

SEC 9: in che modo proteggi i dati in transito?

Proteggi i dati in transito implementando più controlli per ridurre il rischio di accessi non autorizzati o perdita.

AWS offre molteplici mezzi per crittografare i dati a riposo e in transito. Nei nostri servizi integriamo funzionalità che semplificano la crittografia dei dati. Ad esempio, abbiamo implementato la crittografia lato server (SSE) per Amazon S3 per semplificare l'archiviazione dei dati in forma crittografata. È inoltre possibile disporre che l'intero processo di crittografia e decrittografia HTTPS (generalmente noto come terminazione SSL) sia gestito da Elastic Load Balancing (ELB).

Risposta agli incidenti

Anche con controlli preventivi e investigativi estremamente maturi, la tua organizzazione dovrebbe comunque attuare processi per rispondere e mitigare il potenziale impatto di incidenti di sicurezza. L'architettura del carico di lavoro influisce fortemente sulla capacità dei team di operare efficacemente durante un incidente, isolare o contenere sistemi e ripristinare le operazioni a uno stato ottimale noto. La messa in atto degli strumenti e l'accesso prima di un incidente di sicurezza e la pratica sistematica della risposta agli incidenti durante i giorni di attività ti aiuterà a verificare che la tua architettura sia in grado di supportare indagini e ripristini tempestivi.

In AWS, le seguenti pratiche agevolano una risposta efficace agli incidenti:

- Sono disponibili log dettagliati che contengono contenuti importanti, come l'accesso ai file e le modifiche.
- Gli eventi possono essere elaborati automaticamente e possono avviare strumenti che automatizzano le risposte mediante l'uso delle API di AWS.
- Puoi effettuare il pre-provisioning degli strumenti e una "camera bianca" utilizzando AWS CloudFormation. In questo modo puoi effettuare indagini forensi in un ambiente sicuro e isolato.

La seguente domanda si concentra su queste considerazioni relative alla sicurezza.

SEC 10: in che modo prevedi gli incidenti, vi fornisci risposta e recuperi dagli stessi?

La preparazione è cruciale per un esame tempestivo ed efficace degli incidenti di sicurezza, nonché per la risposta e il ripristino, così da ridurre al minimo potenziali interruzioni dell'orga nizzazione.

Verifica di poter garantire rapidamente l'accesso al tuo team addetto alla sicurezza e automatizzare l'isolamento delle istanze, oltre che acquisire i dati e lo stato per le indagini forensi.

Sicurezza delle applicazioni

Il termine sicurezza delle applicazioni (AppSec) descrive il processo complessivo di progettazione, creazione e test delle proprietà di sicurezza dei carichi di lavoro sviluppati. Devi individuare persone sufficientemente qualificate nell'organizzazione, comprendere le proprietà di sicurezza dell'infrastruttura di sviluppo e rilascio e usare l'automazione per identificare i problemi correlati alla sicurezza.

L'adozione di test della sicurezza delle applicazioni come componente regolare del ciclo di vita di sviluppo del software e dei processi successivi al rilascio consente di convalidare la presenza di un meccanismo strutturato per identificare, correggere e prevenire problemi di sicurezza delle applicazioni nell'ambiente di produzione.

La metodologia di sviluppo delle applicazioni deve includere controlli di sicurezza durante la progettazione, l'implementazione e il funzionamento dei carichi di lavoro. Nel frattempo, allinea il processo per una continua riduzione degli errori e l'azzeramento del debito tecnico. Ad esempio, usando la modellazione delle minacce durante la fase di progettazione, puoi individuare i difetti di progettazione e correggerli più facilmente e in modo meno costoso anziché attendere e mitigarli in un secondo momento.

Costi e complessità associati alla correzione dei difetti sono in genere inferiori nelle fasi iniziali del ciclo di vita di sviluppo del software. Il modo più semplice per risolvere i problemi è non averne affatto ed è per questo che un modello di rischio iniziale ti permette di concentrarti sui risultati corretti sin dalla fase di progettazione. Con l'evolvere del programma per la sicurezza delle applicazioni, puoi aumentare la quantità di test eseguiti tramite l'automazione, migliorare l'attendibilità del feedback degli sviluppatori e ridurre il tempo necessario per le revisioni della sicurezza. Tutte queste iniziative migliorano la qualità del software sviluppato e accelerano la distribuzione di funzionalità nell'ambiente di produzione.

Le presenti linee guida per l'implementazione si concentrano su quattro aree: organizzazione e cultura, sicurezza della pipeline, sicurezza nella pipeline e gestione delle dipendenze. Ogni area fornisce un set di principi che puoi implementare e una visione completa di come progettare, sviluppare, compilare, implementare ed eseguire carichi di lavoro.

AWS offre diversi approcci da usare per gestire il programma per la sicurezza delle applicazioni. Alcuni sono basati sulla tecnologia, mentre altri sono incentrati sulle persone e gli aspetti organizzativi del programma.

La seguente domanda si concentra su queste considerazioni relative alla sicurezza dell'applicazione.

SEC 11: come si integrano e convalidano le proprietà di sicurezza delle applicazioni lungo il ciclo di vita di progettazione, sviluppo e implementazione?

La formazione del personale, l'esecuzione di test tramite automazione, l'identificazione delle dipendenze e la convalida delle proprietà di sicurezza di strumenti e applicazioni riducono la probabilità del verificarsi di problemi di sicurezza nei carichi di lavoro di produzione.

Risorse

Consulta le seguenti risorse per ulteriori informazioni sulle best practice relative alla sicurezza.

Documentazione

- AWS Sicurezza del cloud
- AWS Conformità
- AWS Security Blog
- AWS Security Maturity Model

Whitepaper

- Pilastro della sicurezza
- AWS Panoramica della sicurezza
- AWS Rischio e conformità

Video

- AWS Stato della sicurezza del cloud
- Panoramica sulla responsabilità condivisa

Affidabilità

Il pilastro dell'affidabilità comprende la capacità di un carico di lavoro di eseguire la funzione attesa in modo corretto e coerente quando previsto. Ciò comprende la possibilità di utilizzare e testare il

carico di lavoro per tutto il ciclo di vita. Il presente documento fornisce linee guida dettagliate sulle best practice per l'implementazione di carichi di lavoro affidabili in AWS.

Il pilastro dell'affidabilità offre una panoramica dei principi di progettazione, delle best practice e delle domande. Puoi trovare linee guida prescrittive sull'implementazione nel whitepaper sul pilastro dell'affidabilità.

Argomenti

Principi di progettazione

Esistono cinque principi di progettazione per l'affidabilità nel cloud:

- Ripristino automatico in caso di guasto: monitorando un carico di lavoro per gli indicatori chiave di prestazioni (KPI), puoi avviare l'automazione quando una soglia viene superata. Questi KPI dovrebbero essere una misura del valore aziendale, non degli aspetti tecnici del funzionamento del servizio. Ciò consente la notifica e il tracciamento automatici degli errori e i processi di recupero automatizzati che aggirano o riparano l'errore. Con un'automazione più sofisticata, è possibile anticipare e correggere gli errori prima che si verifichino.
- Test delle procedure di ripristino: in un ambiente on-premises, spesso vengono eseguiti test per dimostrare che il carico di lavoro funziona in uno scenario specifico. I test non vengono in genere utilizzati per convalidare le strategie di ripristino. Nel cloud, puoi testare il modo in cui il carico di lavoro incorre nell'errore e convalidare le procedure di ripristino. Puoi utilizzare l'automazione per simulare diversi errori o ricreare scenari che in precedenza hanno portato a errori. Questo approccio presenta percorsi di errore che è possibile testare e correggere prima che si verifichi uno scenario di errore reale, riducendo così il rischio.
- Scalare a livello orizzontale per aumentare la disponibilità dei carichi di lavoro aggregati: sostituisci una risorsa grande con più risorse piccole per ridurre l'impatto di un singolo guasto sul carico di lavoro complessivo. Distribuisci le richieste tra più risorse di dimensioni inferiori per verificare che non condividano un punto di errore comune.
- Smetti di fare ipotesi sulla capacità: una causa comune di guasti nei carichi di lavoro on-premises è la saturazione delle risorse, quando le richieste assegnate a un carico di lavoro superano la

capacità di quel carico di lavoro (questo è spesso l'obiettivo di attacchi di tipo Denial of Service). Nel cloud, è possibile monitorare la domanda e l'utilizzo dei carichi di lavoro, nonché automatizzare l'aggiunta o la rimozione di risorse per mantenere il livello più efficiente, al fine di soddisfare la domanda senza un provisioning eccessivo o inferiore. Esistono ancora limiti, ma alcune quote possono essere controllate e altre possono essere gestite (consulta Gestione di Service Quotas e vincoli).

- Gestione del cambiamento tramite l''automazione: le modifiche all'infrastruttura andrebbero apportate utilizzando l'automazione. Le modifiche che devono essere gestite includono quelle all'automazione, che possono quindi essere monitorate e revisionate.

Definizione

Esistono quattro aree di best practice per l'affidabilità nel cloud:

- Fondamenti
- Architettura del carico di lavoro
- Gestione delle modifiche
- Gestione dei guasti

Per ottenere affidabilità, è necessario iniziare dalle basi: un ambiente in cui Service Quotas e topologia di rete sono in grado di supportare il carico di lavoro. L'architettura del carico di lavoro del sistema distribuito deve essere progettata per prevenire e mitigare i guasti. Il carico di lavoro deve gestire le variazioni nella domanda o nei requisiti e deve essere progettato per rilevare il guasto e applicare autonomamente le correzioni in automatico.

Best practice

Argomenti

Fondamenti

I requisiti di base sono quelli il cui ambito si estende oltre un singolo carico di lavoro o progetto. Prima di progettare qualsiasi sistema, occorre stabilire i requisiti fondamentali che influenzano l'affidabilità. Ad esempio, è necessario disporre di una larghezza di banda della rete sufficiente verso il data center.

Con AWS, la maggior parte di questi requisiti di base è già incorporata o può essere affrontata in base alle esigenze. Il cloud è progettato per essere quasi illimitato, perciò è responsabilità di AWS soddisfare i requisiti di capacità di rete e di elaborazione sufficienti, permettendoti di modificare le dimensioni delle risorse e le allocazioni on demand.

Le seguenti domande si concentrano su queste considerazioni relative all'affidabilità. Per l'elenco completo delle domande e delle best practice relative all'affidabilità, consulta l'Appendice.

REL 1: in che modo gestisci Service Quotas e vincoli?

Per le architetture di carichi di lavoro basate sul cloud, esistono Service Quotas (chiamate anche restrizioni dei servizi). Queste quote sono presenti per evitare di effettuare accidentalmente il provisioning di più risorse di quelle necessarie e limitare i tassi di richiesta sulle operazioni API in modo da proteggere i servizi da un uso illecito. Esistono anche vincoli di risorse, ad esempio la velocità con cui è possibile trasferire i bit su un cavo in fibra ottica o lo spazio di archiviazione su un disco fisico.

REL 2: in che modo pianifichi la topologia di rete?

I carichi di lavoro sono spesso presenti in più ambienti. Questi includono più ambienti cloud (sia accessibili pubblicamente sia privati) e, possibilmente, l'infrastruttura del data center esistente. I piani devono includere considerazioni di rete, ad esempio connettività intrasistema e intersistema, gestione di indirizzi IP pubblici, gestione di indirizzi IP privati e risoluzione dei nomi di dominio.

Architettura del carico di lavoro

Un carico di lavoro affidabile comincia con decisioni iniziali di progettazione sia per il software sia per l'infrastruttura. Le tue scelte architetturali avranno un impatto sul comportamento del carico di

lavoro su tutti i pilastri del Framework Well-Architected. Per l'affidabilità, è necessario seguire modelli specifici.

Con AWS, gli sviluppatori di carichi di lavoro possono scegliere i linguaggi e le tecnologie da utilizzare. AWS Gli SDK semplificano la scrittura di codici fornendo API specifiche dei linguaggi per i servizi AWS. Questi SDK, oltre alla scelta dei linguaggi, permettono agli sviluppatori di implementare le best practice di affidabilità elencate qui. Gli sviluppatori possono anche leggere e scoprire come Amazon crea e gestisce software nella Amazon Builders' Library.

Le seguenti domande si concentrano su queste considerazioni relative all'affidabilità.

REL 3: in che modo progetti l'architettura del servizio di carico di lavoro?

Crea carichi di lavoro altamente scalabili e affidabili utilizzando un'architettura orientata ai servizi (SOA) o un'architettura di microservizi. L'architettura orientata ai servizi (SOA) è la pratica di rendere i componenti software riutilizzabili tramite interfacce di servizio. L'architettura dei microservizi va oltre, per rendere i componenti più piccoli e semplici.

REL 4: in che modo progetti le interazioni in un sistema distribuito per evitare errori?

I sistemi distribuiti si basano sulle reti di comunicazione per interconnettere i componenti, ad esempio server o servizi. Il carico di lavoro deve funzionare in modo affidabile nonostante la perdita o la latenza dei dati su queste reti. I componenti del sistema distribuito devono funzionar e in modo da non influire negativamente su altri componenti o sul carico di lavoro. Queste best practice consentono di prevenire gli errori e migliorare il tempo medio tra guasti (MTBF).

REL 5: in che modo progetti le interazioni in un sistema distribuito per mitigare o affrontare gli errori?

I sistemi distribuiti si basano sulle reti di comunicazione per interconnettere i componenti (ad esempio server o servizi). Il carico di lavoro deve funzionare in modo affidabile nonostante la perdita o la latenza dei dati su queste reti. I componenti del sistema distribuito devono funzionar e in modo da non influire negativamente su altri componenti o sul carico di lavoro. Queste best practice permettono ai carichi di lavoro di tollerare le sollecitazioni o i guasti, recuperare più

REL 5: in che modo progetti le interazioni in un sistema distribuito per mitigare o affrontare gli errori?

rapidamente e mitigare l'impatto di tali problemi. Il risultato è un miglioramento del tempo medio di ripristino (MTTR).

Gestione delle modifiche

Le modifiche apportate al carico di lavoro o al relativo ambiente devono essere previste e gestite in modo da garantire l'affidabilità del carico di lavoro. Certe modifiche al carico di lavoro sono imposte da fattori esterni, quali i picchi di domanda, e anche altre modifiche dipendono da fattori interni, quali le distribuzioni delle funzionalità e le patch di sicurezza.

Utilizzando AWS, puoi monitorare il comportamento di un carico di lavoro e automatizzare la risposta ai KPI. Ad esempio, il carico di lavoro può aggiungere ulteriori server man mano che il carico di lavoro acquisisce più utenti. È possibile controllare chi dispone dell'autorizzazione per apportare modifiche al carico di lavoro ed eseguire l'audit della cronologia di tali modifiche.

Le seguenti domande si concentrano su queste considerazioni relative all'affidabilità.

REL 6: in che modo monitori le risorse del carico di lavoro?

I log e le metriche sono strumenti molto efficaci per ottenere informazioni sullo stato del carico di lavoro. Puoi configurare il carico di lavoro in modo da monitorare i log e le metriche e inviare notifiche in caso di superamento delle soglie o di eventi significativi. Il monitoraggio permette al carico di lavoro di riconoscere il superamento delle soglie di prestazioni basse o il verificarsi di errori, in modo da ripristinarlo in automatico di conseguenza.

REL 7: in che modo progetti il carico di lavoro per adattarti ai cambiamenti della domanda?

Un carico di lavoro scalabile garantisce l'elasticità per aggiungere o rimuovere risorse in automatic o, in modo che sussista una stretta corrispondenza con la domanda attuale in un dato momento.

REL 8: in che modo implementi le modifiche?

Per implementare nuove funzionalità e verificare che i carichi di lavoro e l'ambiente operativo eseguano software noti e che sia possibile applicare patch o sostituirli in modo prevedibile, sono necessarie modifiche controllate. Se invece non sono controllate, risulta difficile prevederne l'effetto o risolvere eventuali problemi che causano.

Progettando un carico di lavoro in grado di aggiungere e rimuovere automaticamente le risorse in risposta ai cambiamenti della domanda, non solo si aumenta l'affidabilità, ma si convalida anche che il successo aziendale non diventi un peso. Con il monitoraggio attivo, il tuo team verrà avvisato automaticamente quando gli indicatori KPI si discostano dalle norme previste. La registrazione automatica delle modifiche al proprio ambiente permette di controllare e identificare rapidamente le azioni che potrebbero avere influito sull'affidabilità. I controlli sulla gestione delle modifiche certificano la possibilità di applicare le regole che garantiscono l'affidabilità di cui hai bisogno.

Gestione dei guasti

In qualsiasi sistema di ragionevole complessità è previsto che si verifichino errori. L'affidabilità richiede che il carico di lavoro venga a conoscenza degli errori nel momento in cui si verificano e intervenga per evitare conseguenze sulla disponibilità. I carichi di lavoro devono essere in grado di affrontare errori e risolvere automaticamente i problemi.

Con AWS, puoi sfruttare l'automazione per reagire ai dati di monitoraggio. Ad esempio, quando un determinato parametro supera una soglia, è possibile avviare un'azione automatizzata per risolvere il problema. Inoltre, anziché tentare di diagnosticare e correggere una risorsa guasta che fa parte del tuo ambiente di produzione, puoi sostituirla con una nuova ed eseguire l'analisi sulla risorsa guasta fuori banda. Poiché il cloud consente di creare versioni temporanee di un intero sistema a basso costo, è possibile utilizzare i test automatizzati per verificare i processi di recupero completi.

Le seguenti domande si concentrano su queste considerazioni relative all'affidabilità.

REL 9: in che modo esegui il backup dei dati?

Esegui il backup dei dati, delle applicazioni e della configurazione per soddisfare i requisiti relativi agli obiettivi del tempo di ripristino (RTO) e agli obiettivi del punto di ripristino (RPO).

REL 10: in che modo utilizzi l'isolamento dei guasti per proteggere il carico di lavoro?

L'isolamento dei guasti limita l'impatto di un guasto di un componente o di un sistema entro una determinata barriera. Con un isolamento adeguato, i componenti al di fuori della barriera non subiscono gli effetti del guasto. Utilizzando più barriere per l'isolamento dei guasti, è possibile rendere un carico di lavoro più resiliente ai guasti.

REL 11: in che modo progetti il carico di lavoro affinché resista ai guasti dei componenti?

I carichi di lavoro con requisiti di disponibilità elevata e MTTR (Mean Time To Recovery) basso devono essere progettati per garantire la resilienza.

REL 12: in che modo testi l'affidabilità?

Dopo aver progettato il carico di lavoro in modo da essere resiliente alle sollecitazioni della produzione, i test sono l'unico modo per verificare il funzionamento corretto e offrire la resilienza prevista.

REL 13: come pianifichi il disaster recovery?

Avere backup e componenti del carico di lavoro ridondanti in loco è l'inizio della strategia di disaster recovery. RTO ed RPO sono gli obiettivi per il ripristino del carico di lavoro. Imposta questi valori in base alle esigenze aziendali. Implementa una strategia per raggiungere questi obiettivi, prendendo in considerazione le posizioni e la funzione delle risorse e dei dati del carico di lavoro. La probabilità di interruzione e il costo del ripristino sono fattori chiave che aiutano a comunicare il valore aziendale che può avere il disaster recovery per un carico di lavoro.

Esegui regolarmente il backup dei dati e testa i file di backup per verificare di poter effettuare il ripristino dopo errori sia logici che fisici. Una chiave per la gestione dei guasti è il test frequente e automatico dei carichi di lavoro che causano gli errori e quindi osservare come si ripristinano. Esegui questa operazione regolarmente e verifica che tali test vengano avviati anche dopo importanti cambiamenti del carico di lavoro. Traccia attivamente i KPI, e anche l'obiettivo del tempo di ripristino (RTO) e l'obiettivo del punto di ripristino (RPO), per valutare la resilienza di un carico di lavoro

(specialmente in scenari di test degli errori). Il monitoraggio dei KPI ti aiuterà a identificare e mitigare i singoli punti di errore. L'obiettivo è testare a fondo i processi di ripristino del carico di lavoro in modo da avere la certezza di poter recuperare tutti i dati e continuare a servire i propri clienti, anche di fronte a problemi prolungati. I processi di recupero dovrebbero essere testati tanto quanto i normali processi di produzione.

Risorse

Consulta le seguenti risorse per ulteriori informazioni sulle best practice per l'affidabilità.

Documentazione

- Documentazione AWS
- Infrastruttura globale di AWS
- AWS Auto Scaling: How Scaling Plans Work
- Che cos'è AWS Backup?

Whitepaper

- Pilastro dell'affidabilità: AWS Well-Architected
- Implementazione di microservizi in AWS

Efficienza delle prestazioni

Il pilastro dell'efficienza delle prestazioni include la capacità di utilizzare in modo efficiente le risorse nel cloud per soddisfare i requisiti in termini di prestazione e di mantenere tale efficienza a fronte al cambiamento della domanda e all'evoluzione delle tecnologie.

Il pilastro dell'efficienza delle prestazioni offre una panoramica dei principi di progettazione, delle best practice e delle domande. Puoi trovare linee guida prescrittive sull'implementazione nel whitepaper sul pilastro dell'efficienza delle prestazioni.

Argomenti

- Principi di progettazione
- Definizione
- Best practice

- Risorse

Principi di progettazione

Esistono cinque principi di progettazione per l'efficienza delle prestazioni nel cloud:

- Estendi a tutti le tecnologie avanzate: agevola l'implementazione di tecnologie avanzate da parte del tuo team delegando le attività complesse al tuo fornitore di cloud. Anziché chiedere al team IT di imparare come adottare e gestire una nuova tecnologia, valuta l'opportunità di utilizzare la tecnologia come servizio. Ad esempio, No SQL database, transcodifica multimediale e apprendimento automatico sono tutte tecnologie che richiedono competenze specialistiche. Nel cloud, tali tecnologie diventano servizi che il tuo team può semplicemente utilizzare mentre si concentra sullo sviluppo di un prodotto invece che sul provisioning e sulla gestione delle risorse.
- Diventa globale in pochi minuti: l'implementazione del carico di lavoro in più AWS regioni del mondo ti consente di fornire una latenza inferiore e un'esperienza migliore ai tuoi clienti a costi minimi.
- Utilizza architetture serverless: scegliendo le architetture serverless, non avrai più bisogno di gestire e mantenere server fisici per portare a termine le attività di elaborazione tradizionali. Ad esempio, i servizi di storage serverless possono agire da siti web statici, eliminando la necessità di server web, mentre i servizi di eventi possono ospitare il codice. Questo elimina l'onere operativo della gestione dei server fisici, con una riduzione dei costi delle transazioni, dal momento che servizi gestiti di questo tipo funzionano a livello di cloud.
- Sperimenta più di frequente: le risorse virtuali e automatizzabili ti permettono di portare a termine velocemente i test comparativi utilizzando diversi tipi di istanze, storage o configurazioni.
- Prendi in considerazione la comprensione meccanica: scopri come vengono consumati i servizi cloud e utilizza sempre l'approccio tecnologico più adatto ai tuoi obiettivi di carico di lavoro. Ad esempio, prendi in considerazione gli schemi di accesso ai dati quando selezioni una strategia basata su database o archiviazione.

Definizione

Esistono cinque aree di best practice per l'efficienza delle prestazioni nel cloud:

- Scelta dell'architettura
- Calcolo e hardware
- Gestione dei dati

- Reti e distribuzione di contenuti
- Processo e cultura

Utilizza un approccio basato sui dati per la creazione di un'architettura a prestazioni elevate. Raccogli dati su tutti gli aspetti dell'architettura, dalla progettazione di alto livello alla selezione e alla configurazione dei tipi di risorse.

La revisione periodica delle tue scelte conferma che stai sfruttando il cloud in continua evoluzione. AWS Il monitoraggio ti assicurerà di essere consapevole di qualsiasi divergenza rispetto alle prestazioni previste. Infine, puoi raggiungere dei compromessi nella tua architettura per migliorare le prestazioni, ad esempio utilizzando la compressione o la memorizzazione nella cache oppure allentando i requisiti di coerenza.

Best practice

Argomenti

Scelta dell'architettura

La soluzione ottimale per un determinato carico di lavoro può variare e le soluzioni spesso combinano molteplici approcci. I carichi di lavoro Well-Architected utilizzano soluzioni multiple e forniscono funzionalità diverse per migliorare le prestazioni.

AWS le risorse sono disponibili in molti tipi e configurazioni, il che rende più facile trovare un approccio che corrisponda strettamente alle proprie esigenze. Inoltre, puoi trovare opzioni che non sono facili da trovare nelle infrastrutture on-premises. Ad esempio, un servizio gestito come Amazon DynamoDB fornisce un database SQL No completamente gestito con latenza di un millisecondo su qualsiasi scala.

Le seguenti domande si concentrano su queste considerazioni relative all'efficienza delle prestazioni. Per l'elenco completo delle domande e delle best practice relative all'efficienza delle prestazioni, consulta l'Appendice.

PERF1: Come selezionate le risorse cloud e i modelli di architettura appropriati per il vostro carico di lavoro?

Spesso sono necessari molteplici approcci per ottenere prestazioni più efficienti in un carico di lavoro. I sistemi Well-Architected utilizzano più soluzioni e funzionalità per migliorare le prestazio ni.

Calcolo e hardware

La soluzione ottimale in termini di calcolo per un determinato carico di lavoro potrebbe variare in base alla progettazione dell'applicazione, ai modelli di utilizzo e alle impostazioni di configurazione. Le architetture possono utilizzare diverse soluzioni di calcolo per vari componenti e impiegare funzionalità diverse per migliorare le prestazioni. Selezionare la soluzione di calcolo sbagliata per un'architettura può ridurre l'efficienza delle prestazioni.

In AWS, il calcolo è disponibile in tre forme: istanze, contenitori e funzioni:

- Le istanze sono server virtualizzati che consentono di modificarne le funzionalità con un pulsante o una chiamata. API Poiché nel cloud le decisioni relative alle risorse non sono cristallizzate nel tempo, è possibile sperimentare vari tipi di server. Attualmente AWS, queste istanze di server virtuali sono disponibili in famiglie e dimensioni diverse e offrono un'ampia varietà di funzionalità, tra cui unità a stato solido (SSDs) e unità di elaborazione grafica (). GPUs
- I container sono un metodo di virtualizzazione del sistema operativo che consente di eseguire un'applicazione e le sue dipendenze in processi con risorse isolate. AWS Fargate è un'elaborazione serverless per contenitori oppure Amazon EC2 può essere utilizzato se hai bisogno di controllare l'installazione, la configurazione e la gestione del tuo ambiente di elaborazione. Puoi anche scegliere tra più piattaforme di orchestrazione dei container: Amazon Elastic Container Service (ECS) o Amazon Elastic Kubernetes Service (). EKS
- Le funzioni astraggono l'ambiente di esecuzione dal codice che desideri eseguire. Ad esempio, AWS Lambda consente di eseguire codice senza eseguire un'istanza.

Le seguenti domande si concentrano su queste considerazioni relative all'efficienza delle prestazioni.

PERF2: Come selezionate e utilizzate le risorse di calcolo nel vostro carico di lavoro?

La soluzione di calcolo più efficiente per un determinato carico di lavoro varia in base alla progettazione dell'applicazione, ai modelli di utilizzo e alle impostazioni di configurazione. Le architetture possono utilizzare diverse soluzioni di elaborazione per vari componenti e attivare funzionalità diverse per migliorare le prestazioni. Selezionare la soluzione di calcolo sbagliata per un'architettura può portare a una riduzione dell'efficienza delle prestazioni.

Gestione dei dati

La soluzione di gestione dei dati ottimale per un particolare sistema varia in base al tipo di dati (blocco, file o oggetto), ai modelli di accesso (casuale o sequenziale), alla velocità effettiva richiesta, alla frequenza di accesso (online, offline, di archiviazione), alla frequenza di aggiornamento (WORMdinamica) e ai vincoli di disponibilità e durabilità. I carichi di lavoro Well-Architected utilizzano archivi dati appositamente progettati che impiegano diverse funzionalità per migliorare le prestazioni.

In AWS, lo storage è disponibile in tre forme: oggetto, blocco e file:

- Lo storage a oggetti fornisce una piattaforma scalabile e durevole per rendere i dati accessibili da qualsiasi posizione Internet per contenuti generati dagli utenti, archivi attivi, computing serverless, storage di big data o backup e ripristino. Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione di oggetti che offre scalabilità, disponibilità dei dati, sicurezza e prestazioni tra le migliori del settore. Amazon S3 è progettato per garantire una durabilità del 99,999999999% (11 9) e memorizza i dati per milioni di applicazioni per aziende in tutto il mondo.
- Lo storage a blocchi offre uno storage a blocchi ad alta disponibilità, coerente e a bassa latenza per ogni host virtuale ed è analogo allo storage collegato direttamente (DAS) o allo Storage Area Network (). SAN Amazon Elastic Block Store (AmazonEBS) è progettato per carichi di lavoro che richiedono uno storage persistente accessibile da EC2 istanze che consente di ottimizzare le applicazioni con la capacità di storage, le prestazioni e i costi corretti.
- Lo storage di file fornisce accesso a un file system condiviso tra più sistemi. Le soluzioni di storage di file come Amazon Elastic File System (AmazonEFS) sono ideali per casi d'uso come archivi di contenuti di grandi dimensioni, ambienti di sviluppo, negozi multimediali o home directory degli utenti. Amazon FSx rende efficiente ed economico il lancio e l'esecuzione dei file system più diffusi in modo da poter sfruttare i ricchi set di funzionalità e le prestazioni rapide dei file system open source e con licenza commerciale ampiamente utilizzati.

Le seguenti domande si concentrano su queste considerazioni relative all'efficienza delle prestazioni.

PERF3: Come archiviate, gestite e accedete ai dati del vostro carico di lavoro?

La soluzione di storage più efficiente per un sistema varia in base al tipo di operazione di accesso (blocco, file o oggetto), ai modelli di accesso (casuale o sequenziale), al throughput richiesto , alla frequenza di accesso (online, offline, di archiviazione), alla frequenza di aggiornamento (WORMdinamica) e ai vincoli di disponibilità e durata. I sistemi Well-Architected utilizzano più soluzioni di storage e attivano funzionalità diverse per migliorare le prestazioni e utilizzare le risorse in modo efficiente.

Reti e distribuzione di contenuti

La soluzione di rete ottimale per un carico di lavoro varia in base a latenza, requisiti di throughput, jitter e larghezza di banda. I vincoli fisici, ad esempio le risorse utente o on-premises, determinano le opzioni di posizione. Questi vincoli possono essere compensati con le posizioni edge o la collocazione delle risorse.

On AWS, la rete è virtualizzata ed è disponibile in diversi tipi e configurazioni. In questo modo è più facile soddisfare le esigenze di rete. AWS offre funzionalità di prodotto (ad esempio Enhanced Networking, istanze ottimizzate per la EC2 rete Amazon, accelerazione del trasferimento Amazon S3 e CloudFront Amazon dinamico) per ottimizzare il traffico di rete. AWS offre anche funzionalità di rete (ad esempio, routing di latenza Amazon Route 53 AWS Direct Connect, VPC endpoint Amazon e AWS Global Accelerator) per ridurre la distanza di rete o il jitter.

Le seguenti domande si concentrano su queste considerazioni relative all'efficienza delle prestazioni.

PERF4: Come selezionate e configurate le risorse di rete nel vostro carico di lavoro?

Questa domanda comprende linee guida e best practice per progettare, configurare e gestire soluzioni di rete e distribuzione di contenuti nel cloud in maniera efficiente.

Processo e cultura

Durante la fase di progettazione dei carichi di lavoro, esistono principi e pratiche che è possibile adottare per gestire al meglio carichi di lavoro cloud efficienti e ad alte prestazioni. Per adottare una

cultura che promuova l'efficienza delle prestazioni dei carichi di lavoro cloud, prendi in considerazione questi principi e pratiche fondamentali.

Per sviluppare questa cultura, considera questi principi chiave:

- Infrastruttura come codice: definisci l'infrastruttura come codice utilizzando approcci come i AWS CloudFormation modelli. L'uso dei modelli ti consente di collocare la tua infrastruttura nel controllo sorgente, insieme al codice e alle configurazioni dell'applicazione. Ciò ti premette di applicare le stesse procedure di sviluppo software all'infrastruttura, in modo da accelerare l'iterazione.
- Pipeline di implementazione: usa una pipeline di integrazione continua/implementazione continua (CI/CD) (ad esempio, repository del codice sorgente, sistemi di sviluppo, distribuzione e automazione dei test) per distribuire la tua infrastruttura. Ciò ti consente di effettuare l'implementazione in modo ripetibile, coerente ed economicamente vantaggioso nel corso dell'iterazione.
- Metriche ben definite: configura e monitora le metriche per acquisire gli indicatori chiave di prestazione (). KPIs Ti consigliamo di adottare parametri tecnici e aziendali. Per i siti Web o le app mobili, le metriche chiave sono l'acquisizione o il rendering. time-to-first-byte Gli altri parametri generalmente validi includono il numero di thread, il tasso di rimozione di oggetti inutili (garbage collection) e gli stati di attesa. I parametri aziendali, come il costo cumulativo aggregato per richiesta, possono indicarti due modi per ridurre i costi. Valuta attentamente il modo in cui prevedi di interpretare i parametri. Ad esempio, potresti scegliere il 99° percentile o quello massimo anziché il valore medio.
- Automatizza i test delle prestazioni: nell'ambito del processo di implementazione, avvia automaticamente i test delle prestazioni dopo che quelli dall'esecuzione più rapida hanno dato esito positivo. L'automazione deve creare un nuovo ambiente, configurare le condizioni iniziali come i dati del test ed eseguire una serie di benchmark e test di carico. I risultati dei test devono essere confrontati con la build, in modo da monitorare le variazioni delle prestazioni nel corso del tempo. Per i test di lunga durata, puoi inserirli nella pipeline in maniera asincrona rispetto al resto della build. In alternativa, puoi eseguire test delle prestazioni durante la notte utilizzando Amazon EC2 Spot Instances.
- Generazione del carico: crea una serie di script di test che replichino i percorsi utente sintetici o pre-registrati. Tali script devono essere idempotenti e non devono essere associati in coppie. Inoltre, potrebbe essere necessario includere script preliminari per garantire risultati validi. Testa gli script il più possibile, per assicurarti che replichino le abitudini di utilizzo in produzione. È possibile utilizzare soluzioni software o software-as-a-service (SaaS) per generare il carico. Valuta se

l'utilizzo delle soluzioni Marketplace AWS e le istanze spot possono essere modi convenienti per generare il carico.

- Visibilità delle prestazioni: i parametri principali devono essere visibili dal team, in particolar modo quelli relativi a ciascuna versione della build. Ciò ti consente di rilevare tendenze positive o negative rilevanti nel corso del tempo. Dovresti anche visualizzare i parametri sul numero di errori o eccezioni per assicurarti di testare un sistema funzionante.
- Visualizzazione: sfrutta le tecniche di visualizzazione che indicano in modo chiaro i punti in cui si verificano problemi di prestazioni, hot spot, stati di attesa o utilizzo ridotto. Sovrapponi i parametri delle prestazioni ai diagrammi architetturali: i grafici delle chiamate o il codice possono aiutarti a individuare più rapidamente i problemi.
- Revisione regolare dei processi: le prestazioni scarse delle architetture sono in genere il risultato di un processo di revisione delle prestazioni inesistente o incompleto. Se la tua architettura offre prestazioni insufficienti, l'implementazione di un processo di revisione delle prestazioni ti consente di favorire il miglioramento delle iterazioni.
- Ottimizzazione continua: adotta una cultura per ottimizzare continuamente l'efficienza delle prestazioni del tuo carico di lavoro cloud.

Le seguenti domande si concentrano su queste considerazioni relative all'efficienza delle prestazioni.

PERF5: Quale processo utilizzate per supportare una maggiore efficienza delle prestazioni per il vostro carico di lavoro?

Durante la fase di progettazione dei carichi di lavoro, esistono principi e pratiche che è possibile adottare per gestire al meglio carichi di lavoro cloud efficienti e ad alte prestazioni. Per adottare una cultura che promuova l'efficienza delle prestazioni dei carichi di lavoro cloud, prendi in considerazione questi principi e pratiche fondamentali.

Risorse

Consulta le seguenti risorse per ulteriori informazioni sulle best practice relative all'efficienza delle prestazioni.

Documentazione

- Ottimizzazione delle prestazioni di Amazon S3

- Prestazioni Amazon EBS Volume

Whitepaper

- Pilastro dell'efficienza delle prestazioni

Video

- AWS re:Invent 2019: EC2 fondamenti di Amazon (-R2) CMP211
- AWS re:Invent 2019: Sessione di leadership: Lo stato di archiviazione dell'unione (01-L) STG2
- AWS re:Invent 2019: Sessione di leadership: database creati appositamente (09-L) AWS DAT2
- AWS re:Invent 2019: Connettività e architetture di rete ibride (-R1) AWSAWS NET317
- AWS re:Invent 2019: Potenziamento di EC2 Amazon di nuova generazione: approfondimenti sul sistema Nitro (03-R2) CMP3
- AWS re:Invent 2019: scalabilità fino ai primi 10 milioni di utenti (-R) ARC211

Ottimizzazione dei costi

Il pilastro dell'ottimizzazione dei costi include la possibilità di eseguire sistemi per offrire valore aggiunto al prezzo più basso.

Il pilastro dell'ottimizzazione dei costi offre una panoramica dei principi di progettazione, delle best practice e delle domande. Le linee guida con le prescrizioni sull'implementazione sono disponibili nel whitepaper sul pilastro dell'ottimizzazione dei costi.

Argomenti

- Principi di progettazione
- Definizione
- Best practice
- Risorse

Principi di progettazione

I principi di progettazione per l'ottimizzazione dei costi nel cloud sono cinque:

- Implementa la gestione finanziaria del cloud: per migliorare i risultati finanziari e accelerare la realizzazione del valore aziendale nel cloud, investi nella gestione finanziaria e nell'ottimizzazione dei costi sul cloud. L'organizzazione deve dedicare tempo e risorse per creare capacità in questo nuovo dominio di gestione della tecnologia e dell'utilizzo. Come per le funzionalità di sicurezza o eccellenza operativa, la capacità si crea tramite lo sviluppo di competenze, programmi, risorse e processi destinati a far diventare un'organizzazione efficiente in termini di costi.
- Adotta un modello a consumo: paga solo le risorse di calcolo che richiedi e incrementa o riduci l'utilizzo a seconda delle reali necessità aziendali anziché sulla base di complesse previsioni. Ad esempio, gli ambienti di test e di sviluppo sono in genere usati solo per otto ore al giorno durante la settimana lavorativa. Puoi interrompere queste risorse quando non le utilizzi, risparmiando potenzialmente il 75% dei costi (40 ore anziché 168).
- Misura l'efficienza complessiva: misura i risultati aziendali del carico di lavoro e i costi associati alla sua fornitura. Usa questi dati per determinare i ricavi che puoi ottenere dal miglioramento dei risultati e dalla riduzione dei costi.
- Smetti di spendere denaro per attività onerose e indifferenziate: AWS si occupa delle attività onerose dei data center come il racking, lo stacking e l'alimentazione dei server. Inoltre, elimina l'onere operativo della gestione di sistemi operativi e applicazioni con servizi gestiti. In questo modo, potrai dedicarti ai clienti e ai progetti aziendali anziché dell'infrastruttura IT.
- Analizza e attribuisci le spese: il cloud semplifica l'individuazione precisa di utilizzo e costo dei sistemi, favorendo l'attribuzione trasparente dei costi IT ai singoli proprietari dei carichi di lavoro. In questo modo puoi misurare il ritorno sull'investimento (ROI), mentre i proprietari dei carichi di lavoro hanno la possibilità di ottimizzare le risorse e ridurre i costi.

Definizione

Esistono cinque aree di best practice per l'ottimizzazione dei costi nel cloud:

- Implementazione della gestione finanziaria del cloud
- Comprensione delle spese e dell'utilizzo
- Risorse convenienti in termini di costo
- Gestione delle risorse di domanda e offerta
- Ottimizzazione nel tempo

Come per gli altri pilastri all'interno del Framework Well-Architected, occorre considerare alcuni compromessi; ad esempio, è meglio ottimizzare la velocità di commercializzazione o i costi? In

alcuni casi, è più efficiente ottimizzare la velocità, per velocizzare il lancio sul mercato, fornire nuove funzionalità o rispettare una scadenza, anziché investire nell'ottimizzazione dei costi iniziali. Talvolta le decisioni di progettazione sono guidate dalla rapidità invece che dai dati, ed esiste sempre la tentazione di sovrascrivere piuttosto che dedicare tempo all'esecuzione di benchmark per la implementazione più conveniente. Questo potrebbe portare a implementazione con provisioning eccessivo e sottoutilizzate. Tuttavia, si tratta di una scelta ragionevole quando devi eseguire il "lift and shift" delle risorse dal tuo ambiente on-premises al cloud e procedere con l'ottimizzazione di conseguenza. Investire in anticipo la giusta quantità di energia in una strategia di ottimizzazione dei costi permette di realizzare i vantaggi economici del cloud in modo più rapido, ottenendo il rispetto costante delle best practice ed evitando un provisioning eccessivo. Le sezioni seguenti forniscono tecniche e best practice per l'implementazione iniziale e continua della gestione finanziaria del cloud e l'ottimizzazione dei costi dei carichi di lavoro.

Best practice

Argomenti

Implementazione della gestione finanziaria del cloud

Con l'adozione del cloud, i team tecnologici innovano più rapidamente grazie a cicli di approvazione, approvvigionamento e implementazione dell'infrastruttura più brevi. Per ottenere valore aggiunto e migliorare gli affari è necessario un nuovo approccio alla gestione finanziaria nel cloud. Questo approccio è la gestione finanziaria del cloud e crea capacità in tutta l'organizzazione implementando competenze, programmi, risorse e processi a livello organizzativo.

Molte organizzazioni sono composte da tante unità con priorità diverse. La capacità di allineare un'organizzazione a un insieme concordato di obiettivi finanziari e di fornire all'organizzazione i meccanismi per raggiungerli permette di creare un'organizzazione più efficiente. Un'organizzazione capace innova e crea più rapidamente, è più agile e si adatta a qualsiasi fattore interno o esterno.

In AWS puoi utilizzare Cost Explorer e, facoltativamente, Amazon Athena e Amazon QuickSight, con il report costi e utilizzo (CUR) per fornire consapevolezza su costi e utilizzo in tutta l'organizzazione.

AWS Budgets fornisce notifiche proattive relative a costi e utilizzo. I blog AWS forniscono informazioni su nuovi servizi e funzionalità così da verificare di essere sempre aggiornati sulle nuove versioni dei servizi.

La seguente domanda si concentra su queste considerazioni relative all'ottimizzazione dei costi. Per l'elenco completo delle domande e delle best practice relative all'ottimizzazione dei costi, consulta l'Appendice.

COST 1: in che modo implementi la gestione finanziaria nel cloud?

L'implementazione della gestione finanziaria del cloud aiuta le organizzazioni a conseguire un valore aggiunto e il successo finanziario ottimizzando i costi e l'utilizzo e dimensionando le risorse in AWS.

Quando crei una funzione di ottimizzazione dei costi, puoi utilizzare i membri e integrare il team con esperti di gestione finanziaria del cloud e ottimizzazione dei costi. I membri già presenti nel team conoscono il funzionamento dell'organizzazione e sono in grado di implementare rapidamente i miglioramenti. Valuta anche la possibilità di includere persone con competenze aggiuntive o specialistiche, ad esempio di analisi e gestione dei progetti.

Quando implementi la consapevolezza dei costi nella tua organizzazione, prova a migliorare o sviluppare i programmi e i processi esistenti. È molto più veloce sviluppare i processi e programmi esistenti, piuttosto che crearne di nuovi. In questo modo puoi ottenere risultati molto più rapidamente.

Comprensione delle spese e dell'utilizzo

La maggiore flessibilità e agilità consentite dal cloud incoraggiano l'innovazione, lo sviluppo e l'implementazione rapidi. Riduce i processi manuali e il tempo associati al provisioning dell'infrastruttura on-premises, tra cui l'identificazione delle specifiche hardware, la negoziazione delle quotazioni dei prezzi, la gestione degli ordini di acquisto, la pianificazione delle spedizioni e la distribuzione delle risorse. Tuttavia, la facilità d'uso e la capacità on demand virtualmente illimitata richiedono un nuovo tipo di mentalità in merito alle spese.

Molte aziende sono caratterizzate da più sistemi gestiti da vari team. La capacità di attribuire i costi delle risorse ai singoli proprietari dell'organizzazione o del prodotto incoraggia un comportamento di utilizzo efficiente e contribuisce a ridurre gli sprechi. L'attribuzione precisa dei costi consente di capire quali prodotti sono effettivamente redditizi e permette anche di prendere decisioni più consapevoli in merito alle destinazioni del budget.

Con AWS puoi creare una struttura di account con AWS Organizations o AWS Control Tower per garantire la separazione e semplificare l'allocazione di costi e utilizzo. Puoi anche utilizzare l'applicazione di tag alle risorse per associare informazioni aziendali e organizzative a utilizzo e costi. Utilizza AWS Cost Explorer per osservare costi e utilizzo, oppure crea analisi e pannelli di controllo personalizzati con Amazon Athena e Amazon QuickSight. Puoi verificare costi e utilizzo con le notifiche di AWS Budgets e controllarli usando AWS Identity and Access Management (IAM) e Service Quotas.

Le seguenti domande si concentrano su queste considerazioni relative all'ottimizzazione dei costi.

COST 2: in che modo gestisci l'utilizzo?

Stabilisci policy e meccanismi per convalidare che i costi sostenuti mentre raggiungi gli obiettivi siano adeguati. Utilizzando un approccio di controllo e bilanciamento reciproco, è possibile innovare senza spendere troppo.

COST 3: in che modo monitori l'utilizzo e il costo?

Stabilisci policy e procedure per monitorare e allocare i costi in modo appropriato. Ciò ti permette di misurare e migliorare l'efficienza in termini di costi del carico di lavoro.

COST 4: in che modo disattivi le risorse?

Implementa il controllo del cambiamento e la gestione delle risorse dall'inizio del progetto alla fine del ciclo di vita. In questo modo sarà più semplice disattivare le risorse inutilizzate per ridurre gli sprechi.

Puoi usare i tag di allocazione dei costi per categorizzare e monitorare il tuo utilizzo di AWS e i costi. Quando applichi dei tag alle tue risorse AWS (come le istanze EC2 o i bucket S3), AWS genera un report su costi e utilizzo con i tuoi tag e i dati sul tuo utilizzo. Puoi applicare tag che rappresentano le categorie di un'organizzazione (come i centri di costo, i nomi dei carichi di lavoro o i proprietari) per organizzare i tuoi costi tra i vari servizi.

Verifica di utilizzare il giusto livello di dettaglio e granularità quando crei report e monitori costi e utilizzo. Per informazioni e tendenze generali, utilizza i dati giornalieri di AWS Cost Explorer. Per

analisi e ispezioni più specifiche, utilizza la granularità oraria di AWS Cost Explorer o Amazon Athena e Amazon Quick con il report costi e utilizzo a granularità oraria.

Associando le risorse taggate al monitoraggio del ciclo di vita dell'entità (dipendenti, progetti), puoi individuare le risorse accantonate o i progetti che non generano più valore per l'organizzazione e devono quindi essere dismessi. Puoi impostare avvisi di fatturazione per ricevere notifiche relative a spese eccessive previste.

Risorse convenienti in termini di costo

Utilizzare risorse e istanze adeguate al tuo carico di lavoro è fondamentale per ridurre i costi. Ad esempio, un processo di creazione di report potrebbe impiegare cinque ore su un server più piccolo, ma un'ora su un server più grande che costa il doppio. Entrambi i server ti offrono lo stesso risultato, ma quello più piccolo comporta un costo più elevato nel tempo.

Un carico di lavoro ben progettato usa le risorse più convenienti, il che può avere un impatto economico positivo e notevole. Hai anche la possibilità di usare i servizi gestiti per ridurre i costi. Ad esempio, invece di mantenere dei server per recapitare le e-mail, puoi usare un servizio che ti invia gli addebiti in base ai messaggi inviati.

Per soddisfare in modo più efficiente le tue necessità, AWS offre un'ampia selezione di offerte flessibili e convenienti per l'acquisto di istanze di Amazon EC2 e altri servizi. Le istanze on demand ti permettono di pagare la capacità di calcolo a ore e non richiedono impegni minimi. Savings Plans e istanze riservare garantiscono risparmi fino al 75% sui prezzi delle istanze on demand. Con le istanze spot, puoi sfruttare la capacità inutilizzata di Amazon EC2 e risparmiare fino al 90% sui prezzi on demand. Le istanze spot risultano adeguate quando il sistema può tollerare l'utilizzo di un parco server in cui i singoli server possano andare e venire dinamicamente, come server Web stateless, elaborazioni batch o quando si usano HPC e Big Data.

Anche la scelta del servizio appropriato può ridurre l'utilizzo e i costi; ad esempio, CloudFront può ridurre al minimo il trasferimento dei dati o ridurre i costi, mentre l'utilizzo di Amazon Aurora su Amazon RDS può eliminare gli elevati costi di licenza dei database.

Le seguenti domande si concentrano su queste considerazioni relative all'ottimizzazione dei costi.

COST 5: in che modo valuti i costi quando selezioni i servizi?

Amazon EC2, Amazon EBS e Amazon S3 sono servizi AWS del blocco predefinito. I servizi gestiti, come Amazon RDS e Amazon DynamoDB, sono servizi AWS di livello superiore o di livello

COST 5: in che modo valuti i costi quando selezioni i servizi?

applicazione. Selezionando i blocchi predefiniti e i servizi gestiti appropriati, è possibile ottimizzare questo carico di lavoro per i costi. Ad esempio, utilizzando i servizi gestiti, puoi ridurre o eliminare gran parte dei costi generali amministrativi e operativi, liberandotene per lavorare su applicazioni e attività correlate al tuo business.

COST 6: in che modo raggiungi gli obiettivi di costo quando selezioni il tipo, le dimensioni e il numero delle risorse?

Assicurati di scegliere la dimensione e il numero delle risorse appropriati per l'attività in questione. Selezionando il tipo, le dimensioni e il numero più convenienti, riduci al minimo gli sprechi.

COST 7: in che modo impieghi i modelli di prezzo per ridurre i costi?

Utilizza il modello di prezzo più appropriato per le tue risorse per ridurre al minimo le spese.

COST 8: in che modo pianifichi i costi per il trasferimento dei dati?

Assicurati di pianificare e monitorare i costi di trasferimento dei dati in modo da poter prendere decisioni sull'architettura per ridurre al minimo i costi. Una modifica piccola ma efficace dell'arch itettura può ridurre drasticamente i costi operativi nel tempo.

Scomponendo i costi durante la selezione del servizio e usando strumenti come Cost Explorer e AWS Trusted Advisor per esaminare con regolarità l'utilizzo di AWS, puoi monitorare attivamente il tuo utilizzo e modificare le implementazioni di conseguenza.

Gestione delle risorse di domanda e offerta

Quando passi al cloud, paghi solo ciò che ti occorre. Puoi fornire risorse in base alla domanda del carico di lavoro nel momento in cui sono necessarie, riducendo così la necessità di un provisioning eccessivo, costoso e dispendioso. Puoi anche gestire la domanda utilizzando tecniche come limitazione (della larghezza di banda della rete), buffering o queuing per allentare la domanda e soddisfarla con meno risorse. In questo modo diminuirai i costi o li posticiperai con un servizio batch.

In AWS puoi predisporre automaticamente le risorse da associare alla domanda di carico di lavoro. Auto Scaling con strategie basate su domanda o tempo ti permette di aggiungere e rimuovere le risorse in base alle esigenze. Se riesci a prevedere le variazioni nella domanda, puoi risparmiare di più e convalidare che le risorse corrispondano alle esigenze del tuo carico di lavoro. Puoi usare Gateway Amazon API per implementare la limitazione (della larghezza di banda della rete) o Amazon SQS per implementare una coda nel tuo carico di lavoro. Entrambi permettono di modificare la richiesta nei componenti del carico di lavoro.

La seguente domanda si concentra su queste considerazioni relative all'ottimizzazione dei costi.

COST 9: come gestisci la domanda e fornisci le risorse?

Per avere un carico di lavoro con costo e prestazioni bilanciate, verifica che venga utilizzato tutto ciò per cui paghi ed evita le istanze molto sottoutilizzate. Un parametro di utilizzo distorto, in qualsiasi delle suddette direzioni, ha un impatto negativo sull'organizzazione, sia per i costi operativi (basse prestazioni a causa di un utilizzo eccessivo) che per le spese inerenti a AWS sprecate (a causa di un provisioning eccessivo).

Quando progetti di modificare le risorse di domanda e offerta, pensa attentamente ai modelli di utilizzo, al tempo necessario per effettuare il provisioning delle nuove risorse e alla prevedibilità del modello di domanda. Quando gestisci la domanda, verifica di disporre di una coda o di un buffer di dimensioni corrette e di rispondere alla domanda del carico di lavoro nel periodo di tempo richiesto.

Ottimizzazione nel tempo

Nel momento in cui AWS rilascia nuovi servizi e funzionalità, è buona prassi rivedere le decisioni esistenti relative all'architettura per verificare che continuino a essere le più convenienti. Man mano che le tue esigenze cambiano, disattiva tempestivamente risorse, interi servizi e sistemi non appena smettono di essere necessari.

L'implementazione di nuove funzionalità o tipi di risorse può ottimizzare il carico di lavoro in modo incrementale e con uno sforzo minimo. In questo modo puoi migliorare continuamente l'efficienza nel tempo e utilizzare le tecnologie più aggiornate per ridurre i costi operativi. Puoi anche sostituire o aggiungere nuovi componenti al carico di lavoro con nuovi servizi. In questo modo puoi aumentare in modo significativo l'efficienza, perciò è essenziale rivedere regolarmente il carico di lavoro e implementare nuovi servizi e caratteristiche.

Le seguenti domande si concentrano su queste considerazioni relative all'ottimizzazione dei costi.

COST 10: in che modo valuti i nuovi servizi?

Nel momento in cui AWS rilascia nuovi servizi e funzionalità, è buona prassi rivedere le decisioni esistenti relative all'architettura per verificare che continuino a essere le più convenienti.

Quando esamini regolarmente le tue implementazioni, valuta in che modo i servizi più recenti possono aiutarti a risparmiare. Ad esempio, Amazon Aurora su Amazon RDS può ridurre i costi dei database relazionali. L'utilizzo di serverless come Lambda consente di eliminare la necessità di utilizzare e gestire le istanze per eseguire il codice.

COST 11: come si valuta il costo dell'impegno?

Valuta il costo dell'impegno delle operazioni nel cloud, rivedi le tue operazioni cloud dispendio se in termini di tempo e automatizzale per ridurre gli sforzi umani e i costi adottando servizi AWS correlati, prodotti di terze parti o strumenti personalizzati.

Risorse

Consulta le seguenti risorse per ulteriori informazioni sulle nostre best practice per l'ottimizzazione dei costi.

Documentazione

- AWS Documentazione di

Whitepaper

- Cost Optimization Pillar

Sostenibilità

Alla base del pilastro della sostenibilità c'è l'attenzione all'impatto ambientale, soprattutto in termini di uso ed efficienza delle fonti energetiche, leve importanti che gli architetti usano per definire interventi diretti mirati a ridurre lo sfruttamento delle risorse. Puoi trovare linee guida prescrittive sull'implementazione nel whitepaper sul pilastro della sostenibilità.

Argomenti

Principi di progettazione

Esistono sei principi di progettazione per la sostenibilità nel cloud:

- Analizza il tuo impatto: misura l'impatto del tuo carico di lavoro cloud e definiscine l'impatto futuro. Nella tua analisi includi ogni fonte di impatto: quelle derivanti dall'uso dei prodotti da parte dei tuoi clienti e quelle derivanti dalla rimozione e dal ritiro finali dal mercato. Confronta l'output di produzione e l'impatto totale dei tuoi carichi di lavoro cloud, partendo dall'analisi di risorse ed emissioni richieste per unità di lavoro. Utilizzate questi dati per stabilire gli indicatori chiave di prestazione (KPIs), valutare i modi per migliorare la produttività riducendo l'impatto e stimare l'impatto delle modifiche proposte nel tempo.
- Stabilisci obiettivi di sostenibilità: per ciascun carico di lavoro cloud, stabilisci obiettivi di sostenibilità a lungo termine, come, ad esempio, ridurre le risorse di calcolo e di archiviazione richieste per ciascuna transazione. Modella il ritorno sugli investimenti finalizzati alle migliorie in materia di sostenibilità per i carichi di lavoro esistenti e offri ai proprietari le risorse di cui hanno bisogno per investire negli obiettivi di sostenibilità. Pianifica lo sviluppo e progetta i tuoi carichi di lavoro in modo che la crescita comporti un impatto meno intenso se misurato rispetto a un'unità appropriata, come l'utente o la transazione. Gli obiettivi ti aiutano ad avvalorare un progetto più ampio di sostenibilità che coinvolge la tua azienda o la tua organizzazione, a identificare le regressioni e a dare la priorità a quelle aree che offrono un maggiore potenziale di miglioramento.
- Massimizza l'utilizzo: dimensiona correttamente i carichi di lavoro e implementa un progetto per verificare un utilizzo elevato e ottimizzare l'efficienza energetica dell'hardware sottostante. Due host in esecuzione con una percentuale di utilizzo pari al 30% sono meno efficienti di un host in esecuzione al 60%, se consideriamo il consumo di base per host. Allo stesso tempo, elimina o riduci le risorse, le elaborazioni e le archiviazioni inattive per ridurre l'energia totale richiesta per alimentare il tuo carico di lavoro.
- Anticipa e adotta nuove offerte hardware e software più efficienti: supporta i miglioramenti a monte apportati dai tuoi partner e fornitori così da ridurre l'impatto dei tuoi carichi di lavoro sul cloud.

Monitora costantemente il mercato e valuta nuove offerte hardware e software più efficienti. Adotta la flessibilità nei tuoi progetti per consentire una rapida adozione di tecnologie nuove ed efficienti.

- Affidati a servizi gestiti: la condivisione dei servizi con un'ampia base clienti consente di ottimizzare l'uso delle risorse e ridurre al tempo stesso l'infrastruttura necessaria per supportare i carichi di lavoro nel cloud. Ad esempio, i clienti possono condividere l'impatto dei componenti comuni dei data center come l'alimentazione e la rete migrando i carichi di lavoro verso Cloud AWS e adottando servizi gestiti, come AWS Fargate per i container serverless, che AWS opera su larga scala ed è responsabile del loro funzionamento efficiente. Utilizza servizi gestiti che possono contribuire a ridurre al minimo l'impatto, come lo spostamento automatico dei dati a cui si accede raramente in cold storage con configurazioni del ciclo di vita di Amazon S3 o Amazon EC2 Auto Scaling per regolare la capacità in base alla domanda.
- Riduci l'impatto a valle dei carichi di lavoro nel cloud: riduci la quantità di energia o di risorse impiegate nell'utilizzo dei tuoi servizi. Riduci la necessità di eseguire aggiornamenti dei tuoi dispositivi per usare i tuoi servizi. Esegui test usando device farm per analizzare l'impatto atteso e conduci altri test con i clienti per capire l'impatto reale derivante dall'uso dei tuoi servizi.

Definizione

Esistono sei aree di best practice per la sostenibilità nel cloud:

- Selezione della regione
- Allineamento alla domanda
- Software e architettura
- Dati
- Hardware e servizi
- Processo e cultura

Sostenibilità nel cloud significa impegnarsi continuamente per ridurre principalmente il consumo di energia e garantire una maggiore efficienza di tutti i componenti di un carico di lavoro, ottenendo il massimo vantaggio dalle risorse allocate e riducendo al minimo le risorse richieste. Tale impegno va dalla selezione iniziale di un linguaggio di programmazione efficace, dall'adozione di algoritmi moderni e dall'uso di tecniche di archiviazione di dati efficienti alla distribuzione in infrastrutture di calcolo valide e correttamente dimensionate e alla riduzione dei requisiti per l'hardware degli utenti finali a potenza elevata.

Best practice

Argomenti

Selezione della regione

La scelta della regione per il carico di lavoro influisce in modo significativo su prestazioniKPIs, costi e impronta di carbonio. Per migliorarliKPIs, dovresti scegliere le regioni per i tuoi carichi di lavoro in base ai requisiti aziendali e agli obiettivi di sostenibilità.

La seguente domanda si concentra su queste considerazioni relative alla sostenibilità. Per l'elenco completo delle domande e delle best practice relative all'affidabilità, consulta l'Appendice.

SUS1: Come selezionate le regioni per il vostro carico di lavoro?

La scelta della regione per il carico di lavoro influisce in modo significativo sul carico di lavoroKPI s, in termini di prestazioni, costi e impronta di carbonio. Per migliorarliKPIs, dovresti scegliere le regioni per i tuoi carichi di lavoro in base ai requisiti aziendali e agli obiettivi di sostenibilità.

Allineamento alla domanda

Il modo in cui gli utenti e le applicazioni utilizzano i tuoi carichi di lavoro e altre risorse può aiutarti a identificare i miglioramenti da implementare per raggiungere gli obiettivi di sostenibilità. Puoi scalare l'infrastruttura in modo che sia costantemente adatta alla domanda e verifica di usare solo le risorse minime necessarie per supportare gli utenti. Allinea i livelli di servizio alle esigenze dei clienti. Colloca le risorse in modo da limitare la rete necessaria per il loro consumo da parte di utenti e applicazioni. Rimuovi gli asset inutilizzati. Offri ai membri del team dispositivi in grado di soddisfarne le esigenze con un impatto minimo in termini di sostenibilità.

La seguente domanda si concentra su queste considerazioni relative alla sostenibilità:

SUS2: Come allineate le risorse cloud alla vostra richiesta?

Il modo in cui gli utenti e le applicazioni utilizzano i tuoi carichi di lavoro e altre risorse può aiutarti a identificare i miglioramenti da implementare per raggiungere gli obiettivi di sostenibilità. Puoi scalare l'infrastruttura in modo che sia costantemente adatta alla domanda e verifica di usare solo le risorse minime necessarie per supportare gli utenti. Allinea i livelli di servizio alle esigenze dei clienti. Colloca le risorse in modo da limitare la rete necessaria per il loro consumo da parte di utenti e applicazioni. Rimuovi gli asset inutilizzati. Offri ai membri del team dispositivi in grado di soddisfarne le esigenze con un impatto minimo in termini di sostenibilità.

Dimensiona l'infrastruttura in base al carico degli utenti: identifica i periodi di utilizzo assente o ridotto e scala le risorse per ridurre capacità in eccesso e migliorare l'efficienza.

Allineamento SLAs agli obiettivi di sostenibilità: definisci e aggiorna gli accordi sui livelli di servizio (SLAs), come la disponibilità o i periodi di conservazione dei dati, per ridurre al minimo il numero di risorse necessarie per supportare il carico di lavoro continuando a soddisfare i requisiti aziendali.

Riduci la creazione e la manutenzione di asset inutilizzati: analizza le risorse delle applicazioni (come report precompilati, set di dati e immagini statiche) e i modelli di accesso alle risorse per identificare ridondanze, sottoutilizzi e obiettivi potenziali di disattivazione. Consolida le risorse generate con contenuti ridondanti (come, ad esempio, report mensili con set di dati e output comuni o in sovrapposizione) per ridurre le risorse utilizzate per la duplicazione degli output. Disattiva le risorse non utilizzate (come, ad esempio, immagini di prodotto non più in vendita) per rilasciare le risorse usate e ridurre il numero di risorse sfruttate per supportare il carico di lavoro.

Ottimizza il posizionamento geografico dei carichi di lavoro in base alle posizioni degli utenti: analizza i modelli di accesso alla rete per capire da quali aree geografiche si connettono i tuoi clienti. Seleziona le regioni e i servizi per ridurre la distanza che il traffico di rete deve percorrere e diminuire così le risorse totali di rete richieste per supportare il tuo carico di lavoro.

Ottimizza le risorse dei membri del team in base alle attività eseguite: ottimizza le risorse fornite ai membri del team per ridurre al minimo l'impatto sulla sostenibilità e supportare al tempo stesso le loro esigenze. Esegui ad esempio operazioni complesse, come rendering e compilazione, su desktop cloud condivisi altamente usati invece che su sistemi per utenti singoli, sottoutilizzati e con un alto dispendio energetico.

Software e architettura

Implementa modelli per eseguire lo smoothing del carico e garantire un utilizzo elevato e coerente delle risorse implementate per ridurre al minimo il loro consumo. In seguito alle modifiche nei comportamenti degli utenti nel tempo, alcuni componenti potrebbero diventare inattivi per mancanza di utilizzo. Rivedi modelli e architetture per consolidare i componenti sottoutilizzati e aumentare l'uso complessivo. Ritira i componenti non più necessari. Analizza le prestazioni dei componenti dei tuoi carichi di lavoro e ottimizza quelli che usano la maggior quantità di risorse. Identifica i dispositivi che i clienti utilizzano per accedere ai servizi e implementa modelli in grado di ridurre al minimo la necessità di aggiornamenti dei dispositivi.

La seguente domanda si concentra su queste considerazioni relative alla sostenibilità:

SUS3: Come sfruttate i modelli di software e architettura per supportare i vostri obiettivi di sostenibilità?

Implementa modelli per eseguire lo smoothing del carico e garantire un utilizzo elevato e coerente delle risorse implementate per ridurre al minimo il loro consumo. In seguito alle modifiche nei comportamenti degli utenti nel tempo, alcuni componenti potrebbero diventare inattivi per mancanza di utilizzo. Rivedi modelli e architetture per consolidare i componenti sottoutilizzati e aumentare l'uso complessivo. Ritira i componenti non più necessari. Analizza le prestazioni dei componenti dei tuoi carichi di lavoro e ottimizza quelli che usano la maggior quantità di risorse. Identifica i dispositivi che i clienti utilizzano per accedere ai servizi e implementa modelli in grado di ridurre al minimo la necessità di aggiornamenti dei dispositivi.

Ottimizza software e architetture per processi asincroni e pianificati: utilizza progettazioni e architetture software efficienti per ridurre al minimo le risorse medie richieste per unità di lavoro. Implementa meccanismi che generano un utilizzo uniforme dei componenti per ridurre le risorse inattive tra le attività e diminuire l'impatto di picchi di carico.

Rimuovi o rifattorizza i componenti dei carichi di lavoro con un utilizzo ridotto o assente: monitora l'attività dei carichi di lavoro per individuare i cambiamenti che si verificano nel tempo nell'utilizzo dei singoli componenti. Elimina i componenti non utilizzati e non più necessari e procedi a rifattorizzare quelli con scarso utilizzo per limitare lo spreco di risorse.

Ottimizza le aree di codice che consumano la maggior parte del tempo o delle risorse: monitora l'attività dei carichi di lavoro per individuare i componenti delle applicazioni che usano la maggior

parte delle risorse. Ottimizza il codice eseguito all'interno di questi componenti per ridurre l'utilizzo delle risorse e massimizzare al tempo stesso le prestazioni.

Ottimizza l'impatto su dispositivi e apparecchiature dei clienti: identifica i dispositivi e le attrezzature che i tuoi clienti usano per accedere ai tuoi servizi, il loro ciclo di vita atteso e l'impatto finanziario e di sostenibilità che deriva dalla loro sostituzione. Implementa modelli e architetture software per ridurre al minimo la necessità dei clienti di sostituire dispositivi e aggiornare attrezzature. Implementa ad esempio nuove caratteristiche usando un codice compatibile con versioni di hardware e sistemi operativi precedenti o gestisci la dimensione dei payload in modo che non superino la capacità di archiviazione del dispositivo target.

Usa i modelli e le architetture software che supportano al meglio l'accesso ai dati e i modelli di archiviazione: scopri come i dati vengono utilizzati all'interno del tuo carico di lavoro, consumati dagli utenti, trasferiti e archiviati. Seleziona tecnologie che ti consentono di ridurre l'elaborazione dei dati e i requisiti di archiviazione.

Gestione dei dati

La seguente domanda si concentra su queste considerazioni relative alla sostenibilità:

SUS4: Come sfruttate le politiche e i modelli di gestione dei dati per supportare i vostri obiettivi di sostenibilità?

Implementa procedure di gestione dei dati per ridurre l'archiviazione allocata richiesta per supportare il carico di lavoro e le risorse necessarie per l'uso correlato. Analizza i tuoi dati e usa tecnologie e configurazioni che supportano nel modo più efficace il valore aziendale dei dati e il modo in cui vengono utilizzati. Esegui il ciclo di vita dei dati su un'archiviazione più efficiente e meno performante al diminuire dei requisiti ed elimina i dati che non sono più necessari.

Implementa una policy di classificazione dei dati: classifica i dati per comprenderne il significato in favore dei risultati aziendali. Usa queste informazioni per stabilire quando trasferire i dati in un'archiviazione più efficiente dal punto di vista energetico o eliminarli in totale sicurezza.

Utilizza tecnologie che supportano l'accesso ai dati e i modelli di archiviazione: usa l'archiviazione in grado di supportare nel modo più efficiente il modo in cui viene effettuato l'accesso ai dati e come vengono archiviati per ridurre la quantità di risorse allocate e supportare al tempo stesso il tuo carico di lavoro. Ad esempio, i dispositivi a stato solido (SSDs) consumano più energia rispetto alle unità

magnetiche e devono essere utilizzati solo per casi di utilizzo attivo dei dati. Usa storage di classe di archiviazione ad alta efficienza energetica per i dati ad accesso infrequente.

Utilizza le policy del ciclo di vita per eliminare i dati non necessari: gestisci il ciclo di vita di tutti i tuoi dati e applica in automatico cronologie di eliminazione per ridurre i requisiti totali di archiviazione del tuo carico di lavoro.

Riduci il provisioning eccessivo nell'archiviazione a blocchi: per ridurre la quantità totale di archiviazione assegnata, crea un'archiviazione a blocchi con l'allocazione di dimensioni in base al carico di lavoro. Usa i volumi elastici per espandere l'archiviazione all'aumentare dei dati senza dover ridimensionare l'archiviazione collegata alle risorse di calcolo. Esamina regolarmente i volumi elastici e riduci i volumi con un provisioning eccessivo per adattarli alla dimensione corrente dei dati.

Elimina i dati ridondanti o non necessari: duplica i dati solo quando è necessario per ridurre la quantità totale di archiviazione utilizzata. Utilizza tecnologie di backup che deduplicano i dati a livello di file e blocco. Limita l'uso di configurazioni Redundant Array of Independent Drives (RAID), tranne laddove richiesto per soddisfare i requisiti. SLAs

Utilizza file system condivisi o archiviazione di oggetti per accedere a dati comuni: adotta l'archiviazione condivisa e singole fonti di verità per evitare la duplicazione dei dati e ridurre i requisiti di archiviazione complessiva del tuo carico di lavoro. Recupera i dati dall'archiviazione condivisa solo in base alle esigenze. Distacca volumi non utilizzati per rilasciare le risorse. Riduci al minimo gli spostamenti dei dati tra le reti: usa un'archiviazione condivisa e accedi ai dati da archivi regionali per contenere le risorse di rete totali necessarie per supportare i trasferimenti dei dati per il carico di lavoro.

Esegui il backup dei dati solo quando sono difficili da ricreare: per ridurre al minimo l'uso delle risorse di archiviazione, esegui il backup solo dei dati che abbiano un valore aziendale o siano considerati necessari per soddisfare requisiti di conformità. Esamina le policy di backup ed escludi l'archiviazione temporanea che non offre valore in uno scenario di ripristino.

Hardware e servizi

Cerca opportunità per ridurre l'impatto dei carichi di lavoro in termini di sostenibilità apportando modifiche alle tue pratiche di gestione hardware. Riduci al minimo la quantità di hardware necessaria per il provisioning e l'implementazione e scegli l'hardware e i servizi più efficienti per il singolo carico di lavoro.

La seguente domanda si concentra su queste considerazioni relative alla sostenibilità:

SUS5: Come selezionate e utilizzate l'hardware e i servizi cloud nella vostra architettura per supportare i vostri obiettivi di sostenibilità?

Cerca opportunità per ridurre l'impatto dei carichi di lavoro in termini di sostenibilità apportand o modifiche alle tue pratiche di gestione hardware. Riduci al minimo la quantità di hardware necessaria per il provisioning e l'implementazione e scegli l'hardware e i servizi più efficienti per il singolo carico di lavoro.

Utilizza la quantità minima di hardware per soddisfare le tue esigenze: le funzionalità del cloud consentono di apportare modifiche frequenti alle implementazioni dei carichi di lavoro. Aggiorna i componenti distribuiti man mano che le tue esigenze cambiano.

Usa tipi di istanze con il minimo impatto: monitora costantemente il rilascio di nuovi tipi di istanza e sfrutta le migliorie in tema di efficienza energetica, inclusi i tipi di istanza progettati per supportare carichi di lavoro specifici, come la formazione del machine learning, le inferenze e la transcodifica dei video.

Utilizza i servizi gestiti: i servizi gestiti trasferiscono la responsabilità del mantenimento di un utilizzo medio elevato e dell'ottimizzazione della sostenibilità dell'hardware distribuito su. AWS Utilizza i servizi gestiti per distribuire l'impatto della sostenibilità dei servizi su tutti i tenant relativi, riducendo così il singolo contributo.

Ottimizza l'uso diGPUs: Le unità di elaborazione grafica (GPUs) possono essere una fonte di elevato consumo energetico e molti GPU carichi di lavoro sono estremamente variabili, come il rendering, la transcodifica e la formazione e la modellazione tramite apprendimento automatico. Esegui GPUs le istanze solo per il tempo necessario e disattivale con l'automazione quando non è necessario per ridurre al minimo il consumo di risorse.

Processo e cultura

Cerca opportunità per ridurre l'impatto di sostenibilità apportando modifiche alle tue prassi di sviluppo, test e implementazione.

La seguente domanda si concentra su queste considerazioni relative alla sostenibilità:

SUS6: In che modo i vostri processi organizzativi supportano i vostri obiettivi di sostenibilità?

Cerca opportunità per ridurre l'impatto di sostenibilità apportando modifiche alle tue prassi di sviluppo, test e implementazione.

Adotta operazioni che consentono di integrare rapidamente i miglioramenti orientati alla sostenibilità: testa e convalida potenziali miglioramenti prima di distribuirli in produzione. Tieni in considerazione il costo dei test quando calcoli il potenziale vantaggio futuro di un miglioramento. Sviluppa operazioni di test a basso costo per agevolare la distribuzione di piccoli miglioramenti.

Mantieni aggiornato il tuo carico di lavoro: i sistemi Up-to-date operativi, le librerie e le applicazioni possono migliorare l'efficienza del carico di lavoro e favorire l'adozione di tecnologie più efficienti. Up-to-dateil software potrebbe anche includere funzionalità per misurare l'impatto sulla sostenibilità del carico di lavoro in modo più accurato, poiché i fornitori forniscono funzionalità per raggiungere i propri obiettivi di sostenibilità.

Incrementa l'utilizzo degli ambienti di sviluppo: utilizza l'automazione e il modello Infrastructure as code per rendere operativi gli ambienti di preproduzione quando necessario e dismetterli quando non vengono utilizzati. Un modello comune consiste nel pianificare periodi di disponibilità che coincidano con l'orario di lavoro dei membri del team incaricati dello sviluppo. L'ibernazione è uno strumento utile per preservare lo stato e portare rapidamente le istanze online solo quando necessario. Utilizza tipi di istanze con capacità di espansione, istanze spot, servizi di database elastici, container e altre tecnologie per allineare la capacità di sviluppo e test all'uso.

Utilizza device farm gestite per i test: le device farm gestiste distribuiscono l'impatto di sostenibilità della produzione di hardware e dell'utilizzo delle risorse su più tenant. Le device farm gestite offrono diversi tipi di dispositivi in modo da supportare hardware meno diffusi e di generazioni precedenti e da evitare l'impatto sulla sostenibilità dei clienti dovuti ad aggiornamenti dei dispositivi non necessari.

Risorse

Consulta le seguenti risorse per ulteriori informazioni sulle best practice per la sostenibilità.

Whitepaper

- Pilastro della sostenibilità

Video

- The Climate Pledge

Il processo di revisione

La revisione delle architetture va eseguita in modo coerente, con un approccio che non colpevolizza nessuno, ma che incoraggia ad approfondire gli argomenti. Dovrebbe essere un processo leggero (di ore, non di giorni) più simile a una conversazione che non a un audit. Lo scopo della revisione di un'architettura è identificare dei problemi critici da affrontare o aree di miglioramento. Il risultato della revisione è un insieme di azioni volte a migliorare l'esperienza di utilizzo del carico di lavoro del cliente.

Come discusso nella sezione "Architettura", ogni membro del team deve prendersi la responsabilità della qualità della sua architettura. Consigliamo che i membri del team che hanno sviluppato l'architettura usino il Framework Well-Architected per eseguire costantemente la revisione della loro architettura, piuttosto che fare una riunione di revisione formale. Un approccio quasi continuo permette ai membri del team di aggiornare le risposte man mano che l'architettura evolve e migliorare l'architettura di pari passo alle funzionalità.

Il AWS Well-Architected Framework è allineato al modo in cui esamina sistemi e servizi AWS internamente. Si basa su una serie di principi di progettazione che influenzano l'approccio architettonico e su domande volte a verificare che le persone non trascurino le aree spesso presenti in Root Cause Analysis (). RCA Ogni volta che si verifica un problema significativo con un sistema, un AWS servizio o un cliente interno, lo esaminiamo RCA per vedere se possiamo migliorare i processi di revisione che utilizziamo.

Le revisioni vanno applicate nelle tappe fondamentali del ciclo di vita del prodotto, nelle prime fasi di progettazione per evitare porte a un senso difficili da cambiare e prima della data di lancio. Molte decisioni sono porte reversibili a doppio senso e possono utilizzare un processo leggero. Le porte a un senso sono difficili o impossibili da invertire e richiedono ulteriori ispezioni prima della loro realizzazione. Una volta entrato in produzione, il carico di lavoro continuerà ad evolversi man mano che si aggiungono nuove funzionalità e si modificano le implementazioni tecnologiche. L'architettura del carico di lavoro cambia nel tempo. Devi seguire le best practice di igiene informatica per interrompere il degrado delle caratteristiche man mano che fai evolvere l'architettura. Man mano che l'architettura cambia, dovresti seguire un insieme di processi di igiene informatica tra cui la revisione Well-Architected.

Se vuoi utilizzare la revisione come snapshot una tantum o misura indipendente, dovrai verificare che alla conversazione partecipino tutte le persone appropriate. Spesso ci rendiamo conto che le revisioni sono il primo momento in cui il team comprende per davvero quello che ha implementato.

Un approccio che funziona bene per la revisione dei carichi di lavoro di un altro team consiste in una serie di conversazioni informali sull'architettura in cui ottenere le risposte alla maggior parte delle domande. Quindi puoi fare una o due riunioni di follow up in cui puoi fare chiarezza o approfondire le aree ambigue e il rischio percepito.

Ecco alcuni elementi suggeriti per le tue riunioni:

- Una sala riunioni con una lavagna
- Le stampe di tutti i grafici o delle note di progettazione
- Elenco di azioni a cui è necessaria una out-of-band ricerca per rispondere (ad esempio, «abbiamo attivato la crittografia o no?»)

Dopo avere completato la revisione, dovresti avere un elenco di problemi a cui assegnare delle priorità sulla base del contesto aziendale. Dovrai anche tenere conto dell'impatto di tali problemi sul day-to-day lavoro del tuo team. Se affronti questi problemi in anticipo puoi liberare del tempo per lavorare sulla creazione di valore aziendale anziché dedicarlo a risolvere i problemi ricorrenti. Man mano che affronti i problemi, puoi aggiornare la revisione per vedere in che modo l'architettura sta migliorando.

Il valore di una revisione è evidente dopo averne eseguita una, ma all'inizio un nuovo team potrebbe essere contrario. Ecco alcune obiezioni da gestire per istruire il team sui vantaggi di una revisione:

- "Siamo troppo occupati!" Spesso si sente questa frase quando il team si sta preparando a un grande lancio.
  - Se ti stai preparando per un grande lancio, desidererai che tutto vada bene. La revisione ti permetterà di individuare qualsiasi problema che potresti esserti perso.
  - Ti raccomandiamo di eseguire le revisioni all'inizio del ciclo di vita del prodotto per scoprire i rischi e sviluppare un piano di mitigazione allineato con la roadmap delle funzionalità.
- "Non abbiamo tempo per fare nulla per i risultati!" Spesso questo viene detto quando c'è un evento fisso, come il Super Bowl, di cui si sta occupando il team.
  - Questi eventi non possono essere spostati. Vuoi davvero affrontare l'evento senza conoscere i rischi della tua architettura? Anche se non ti occupi di tutti i problemi in questione, puoi comunque disporre di playbook per affrontarli se si dovessero presentare.
- "Non vogliamo che altri scoprano i segreti della nostra implementazione di soluzioni!"
  - Se poni le domande del Framework Well-Architected, il team noterà che nessuna di esse rivela informazioni proprietarie commerciali o tecniche.

Eseguendo più revisioni con i team della tua organizzazione, potresti identificare delle aree tematiche. Ad esempio, potresti notare che un gruppo di team ha gruppi di problemi in un pilastro o un argomento specifico. Puoi gestire tutte le tue revisioni in modo olistico e identificare tutti i meccanismi, la formazione o le riunioni con gli ingegneri responsabili che possono aiutare a risolvere i problemi tematici.

Conclusioni

Il AWS Well-Architected Framework fornisce le migliori pratiche architettoniche attraverso i sei pilastri per la progettazione e la gestione di sistemi affidabili, sicuri, efficienti, convenienti e sostenibili nel cloud. Il Framework fornisce un insieme di domande che ti permettono di eseguire la revisione di un'architettura esistente o proposta. Fornisce inoltre una serie di AWS best practice per ogni pilastro. L'utilizzo del Framework nella tua architettura ti aiuta a produrre sistemi stabili ed efficienti, che ti permettono di concentrarti sui tuoi requisiti funzionali.

Collaboratori

Le seguenti persone e organizzazioni hanno contribuito a questo documento:

- Brian Carlson, Operations Lead Well-Architected, Amazon Web Services
- Ben Potter, Security Lead Well-Architected, Amazon Web Services
- Seth Eliot, Reliability Lead Well-Architected, Amazon Web Services
- Eric Pullen, Sr. Solutions Architect, Amazon Web Services
- Rodney Lester, Principal Solutions Architect, Amazon Web Services
- Jon Steele, Sr. Technical Account Manager, Amazon Web Services
- Max Ramsay, Principal Security Solutions Architect, Amazon Web Services
- Callum Hughes, Solutions Architect, Amazon Web Services
- Ben Mergen, Senior Cost Lead Solutions Architect, Amazon Web Services
- Chris Kozlowski, Senior Specialist Technical Account Manager, Enterprise Support, Amazon Web Services
- Alex Livingstone, Principal Specialist Solutions Architect, Cloud Operations, Amazon Web Services
- Paul Moran, Principal Technologist, Enterprise Support, Amazon Web Services
- Peter Mullen, Advisory Consultant, Professional Services, Amazon Web Services
- Chris Pates, Senior Specialist Technical Account Manager, Enterprise Support, Amazon Web Services
- Arvind Raghunathan, Principal Specialist Technical Account Manager, Enterprise Support, Amazon Web Services
- Sam Mokhtari, Senior Efficiency Lead Solutions Architect, Amazon Web Services

Approfondimenti

AWS Architecture Center

Conformità di sicurezza nel cloud AWS

AWS Programma Well-Architected Partner

AWS Well-Architected Tool

AWS Homepage Well-Architected

Whitepaper sul pilastro dell'eccellenza operativa

Whitepaper sul pilastro della sicurezza

Whitepaper sul pilastro dell'affidabilità

Whitepaper sul pilastro dell'efficienza delle prestazioni

Whitepaper sul pilastro dell'ottimizzazione dei costi

Whitepaper sul pilastro della sostenibilità

Amazon Builders' Library

Revisioni del documento

Per ricevere una notifica sugli aggiornamenti del presente whitepaper, iscriviti al feed RSS.

Modifica	Descrizione	Data
Aggiornamento principale	Le best practice sono state aggiornate con nuove indicazioni in materia di affidabilità, sicurezza , eccellenza operativa, sostenibilità ed efficienza delle prestazioni. Il pilastro dell'affi dabilità è stato oggetto di una revisione su larga scala e di un aggiornamento di molte best practice. La guida alla sicurezza e all'eccellenza operativa è stata aggiornat a e perfezionata con nuovi servizi e suggerimenti di IA generativa. La sostenibilità ha ricevuto diversi aggiornamenti basati sui servizi AWS e su una nuova best practice.	6 novembre 2024
Aggiornamento principale	In tutti i pilastri sono stati apportati aggiornamenti su larga scala in merito alle best practice. Sono state predispos te best practice correlate a sicurezza e costi.	27 giugno 2024
Aggiornamento principale	Principali aggiornamenti dei pilastri.	3 ottobre 2023

Aggiornamento principale	Best practice aggiornate con prontuario e nuove best practice aggiunte. Nuove domande aggiunte sui pilastri di sicurezza e di ottimizzazione dei costi.	10 aprile 2023
Aggiornamento secondario	Aggiunta della definizione di livello di impegno e aggiornamento delle best practice nell'appendice.	20 ottobre 2022
Aggiornamento principale	Aggiunta del pilastro della sostenibilità e collegamenti aggiornati.	2 dicembre 2021
Aggiornamento principale	Aggiunta al framework del pilastro della sostenibilità.	20 novembre 2021
Aggiornamento secondario	Rimozione del linguaggio non inclusivo.	22 aprile 2021
Aggiornamento secondario	Correzione di diversi collegamenti.	10 marzo 2021
Aggiornamento secondario	Modifiche editoriali di minore entità in varie parti del documento.	15 luglio 2020
Aggiornamento principale	Revisione e riscrittura della maggior parte delle domande e delle risposte.	8 luglio 2020

Aggiornamento del whitepaper	Aggiunta di AWS Well-Arch itected Tool, collegamenti a AWS Well-Architected Labs e ai partner AWS Well-Arch itected, correzioni minori per abilitare la versione in più lingue del framework.	1° luglio 2019
Aggiornamento del whitepaper	Revisione e riscrittura di molte domande e risposte per garantire che le domande si concentrino su un argomento alla volta. Per questo motivo, alcune delle domande precedenti sono state divise in più domande. Aggiunta di termini comuni alle definizioni (carichi di lavoro, component i, ecc.). Presentazione delle domande modificata per includere il testo descrittivo.	1° novembre 2018
Aggiornamento del whitepaper	Aggiornamenti volti a semplific are il testo delle domande e a migliorare la leggibilità.	1° giugno 2018
Aggiornamento del whitepaper	Eccellenza operativa spostata all'inizio della sezione sui pilastri e riscritta in modo che inquadri gli altri pilastri. Aggiornamenti degli altri principi per riflettere l'evoluzi one di AWS.	1° novembre 2017

Aggiornamento del whitepaper	Framework aggiornato per includere i pilastri dell'eccellenza operativa; altri pilastri rivisti e aggiornati per ridurre la duplicazione e incorporare le nozioni apprese grazie alle revisioni eseguite con migliaia di clienti.	1° novembre 2016
Aggiornamenti minori	Appendice aggiornata con le informazioni attuali su Amazon CloudWatch Logs.	1° novembre 2015
Pubblicazione iniziale	Pubblicazione del Framework AWS Well-Architected.	1° ottobre 2015

> **Note**
>
> Per iscriverti agli aggiornamenti RSS, abilita un plug-in RSS nel browser in uso.

Versione del Framework

- 2024-06-27
- 2023-10-03
- 2023-04-10
- 2022-03-31

Appendice: domande e best practice

Questa appendice riassume tutte le domande e le best practice nel Framework AWS Well-Architected.

Pilastri

- Eccellenza operativa
- Sicurezza
- Affidabilità
- Efficienza delle prestazioni
- Ottimizzazione dei costi
- Sostenibilità

Eccellenza operativa

L'eccellenza operativa è un impegno a sviluppare correttamente il software garantendo costantemente un'esperienza cliente di alto livello. Il pilastro dell'eccellenza operativa contiene best practice per organizzare il team, progettare il carico di lavoro, farlo funzionare su scala e seguire la sua evoluzione nel tempo. Puoi trovare linee guida prescrittive sull'implementazione nel whitepaper sul pilastro dell'eccellenza operativa.

Aree delle best practice

- Organizzazione
- Preparazione
- Gestione
- Evoluzione

Organizzazione

Questions

- OPS 1. In che modo stabilisci quali sono le tue priorità?
- OPS 2. Come strutturare la tua organizzazione per supportare i risultati aziendali?
- OPS 3. In che modo la cultura aziendale supporta i risultati aziendali?

OPS 1. In che modo stabilisci quali sono le tue priorità?

È necessario che ognuno comprenda il proprio ruolo per rendere possibile il successo aziendale. Devi disporre di obiettivi comuni al fine di stabilire le priorità per le risorse. Ciò massimizzerà i risultati dei tuoi sforzi.

Best practice

- OPS01-BP01 Valutazione delle esigenze dei clienti esterni
- OPS01-BP02 Valuta le esigenze interne dei clienti
- OPS01-BP03 Valuta i requisiti di governance
- OPS01-BP04 Valutazione dei requisiti di conformità
- OPS01-BP05 Valuta il panorama delle minacce
- OPS01-BP06 Valutazione dei compromessi gestendo vantaggi e rischi

OPS01-BP01 Valutazione delle esigenze dei clienti esterni

Coinvolgi le principali parti interessate, compresi i team aziendali, di sviluppo e operativi, per determinare dove concentrare gli sforzi in base alle esigenze dei clienti esterni. Avrai così una conoscenza approfondita del supporto operativo necessario per raggiungere i risultati aziendali desiderati.

Risultato desiderato:

- Lavori a ritroso partendo dalle esigenze dei clienti.
- Comprendi in che modo le procedure operative supportano i risultati e gli obiettivi aziendali.
- Coinvolgi tutte le parti interessate.
- Disponi di meccanismi per soddisfare le esigenze dei clienti esterni.

Anti-pattern comuni:

- Hai deciso di non fornire il servizio clienti al di fuori dell'orario lavorativo di base, ma non hai esaminato i dati cronologici riguardanti le richieste di supporto. Non sai se questo determinerà un impatto sui tuoi clienti.
- Stai sviluppando una nuova funzionalità, ma non hai coinvolto i clienti per capire se è desiderata ed eventualmente in quale forma; inoltre non hai condotto attività di sperimentazione per convalidarne la necessità e il metodo di distribuzione.

Vantaggi dell'adozione di questa best practice: i clienti le cui esigenze sono soddisfatte hanno maggiori probabilità di rimanere clienti. Valutando e comprendendo le esigenze dei clienti esterni sarà possibile organizzare le attività in base alle priorità e offrire valore aggiunto.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Comprendi le esigenze aziendali: il successo di un'azienda nasce dalla condivisione di obiettivi e conoscenze tra le parti interessate, compresi i team aziendali, di sviluppo e operativi.

Esamina gli obiettivi aziendali, le esigenze e le priorità dei clienti esterni: coinvolgi le principali parti interessate, compresi i team aziendali, di sviluppo e operativi, per discutere obiettivi, esigenze e priorità dei clienti esterni. In tal modo otterrai una conoscenza approfondita del supporto operativo necessario per raggiungere i risultati aziendali e del cliente.

Stabilisci una comprensione condivisa: stabilisci una comprensione condivisa delle funzioni aziendali del carico di lavoro, dei ruoli di ciascuno dei team nel gestire il carico di lavoro e di come questi supportino i tuoi obiettivi aziendali condivisi tra clienti interni ed esterni.

Risorse

Best practice correlate:

- OPS11-BP03 Implementazione di cicli di feedback

OPS01-BP02 Valuta le esigenze interne dei clienti

Coinvolgi le principali parti interessate, compresi i team aziendali, di sviluppo e operativi, nel determinare dove concentrare le attività in base alle esigenze dei clienti interni. Questo ti garantirà una conoscenza approfondita del supporto operativo necessario per raggiungere i risultati aziendali.

Risultato desiderato:

- Utilizzi le priorità definite per concentrare le iniziative di miglioramento delle operazioni laddove avranno il maggiore impatto (ad esempio, sviluppare le competenze dei team, migliorare le prestazioni del carico di lavoro, ridurre i costi, automatizzare i runbook o potenziare il monitoraggio).
- Aggiorni le priorità al mutare delle esigenze.

Anti-pattern comuni:

- Per semplificare la gestione della rete hai deciso di modificare l'assegnazione degli indirizzi IP per i team di prodotto senza consultarli. Non conosci l'impatto che questo avrà sui tuoi team di prodotto.
- Stai implementando un nuovo strumento di sviluppo, ma non hai coinvolto i clienti interni per scoprire se è necessario o se è compatibile con le loro pratiche esistenti.
- Stai implementando un nuovo sistema di monitoraggio, ma non hai contattato i clienti interni per scoprire se hanno esigenze di monitoraggio o reporting da tenere in considerazione.

Vantaggi dell'adozione di questa best practice: valutando e comprendendo le esigenze dei clienti interni consente di organizzare le attività in base a priorità e offrire valore aggiunto.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

- Comprendi le esigenze aziendali: il successo dell'azienda nasce dalla condivisione di obiettivi e conoscenze tra le parti interessate, compresi i team aziendali, di sviluppo e operativi.
- Esamina gli obiettivi aziendali, le esigenze e le priorità dei clienti interni: coinvolgi le principali parti interessate, compresi i team aziendali, di sviluppo e operativi, per discutere obiettivi, esigenze e priorità dei clienti interni. In tal modo otterrai una conoscenza approfondita del supporto operativo necessario per raggiungere i risultati aziendali e del cliente.
- Stabilisci una comprensione condivisa: stabilisci una comprensione condivisa delle funzioni aziendali del carico di lavoro, dei ruoli di ciascuno dei team nel gestire il carico di lavoro e di come questi supportino gli obiettivi aziendali condivisi tra clienti interni ed esterni.

Risorse

Best practice correlate:

- OPS11-BP03 Implementa cicli di feedback

OPS01-BP03 Valuta i requisiti di governance

Con governance si intende l'insieme di policy, regole o framework che un'azienda usa per raggiungere i propri obiettivi. I requisiti di governance vengono generati all'intero dell'organizzazione. Possono influire sui tipi di tecnologia che scegli o sul modo in cui esegui il tuo carico di lavoro. Integra

i requisiti di governance della tua organizzazione nel tuo carico di lavoro. La conformità è la capacità di dimostrare che hai implementato i requisiti di governance.

Risultato desiderato:

- I requisiti di governance sono integrati nel progetto architetturale e nell'operatività del tuo carico di lavoro.
- Puoi dimostrare di aver seguito i requisiti di governance.
- I requisiti di governance vengono rivisti e aggiornati con regolarità.

Anti-pattern comuni:

- La tua azienda richiede che l'account root abbia l'autenticazione multi-fattore. Non sei riuscito a implementare questo requisito e l'account root è compromesso.
- Durante la progettazione del carico di lavoro hai scelto un tipo di istanza non approvata dal dipartimento IT. Non riesci ad avviare il tuo carico di lavoro e devi procedere a una nuova progettazione.
- Devi avere un piano di ripristino di emergenza. Non ne hai uno e il tuo carico di lavoro è vittima di un'interruzione prolungata.
- Il tuo team vuole usare nuove istanze, ma i requisiti di governance non sono stati aggiornati e pertanto non sono consentite.

Vantaggi dell'adozione di questa best practice:

- Rispettare i requisiti di governance permette di allineare il carico di lavoro a policy organizzative di più ampio respiro.
- I requisiti di governance si basano su standard e best practice di settore per la tua organizzazione.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Identifica il requisito di governance collaborando con le parti interessate e le organizzazioni preposte. Includi i requisiti di governance nel tuo carico di lavoro. Dimostra di aver seguito i requisiti di governance.

Esempio del cliente

In AnyCompany Retail, il team operativo del cloud collabora con le parti interessate di tutta l'organizzazione per sviluppare requisiti di governance. Ad esempio, vietano l'SSHaccesso alle EC2 istanze Amazon. Se i team hanno necessità di accedere ai sistemi, devono usare AWS Systems Manager Session Manager. Il team operativo nell'ambiente cloud aggiorna con regolarità i requisiti di governance nel momento in cui vengono rilasciati nuovi servizi.

Passaggi dell'implementazione

1. Identifica le parti interessate per il tuo carico di lavoro, inclusi eventuali team centralizzati.
2. Collabora con le parti interessate per identificare i requisiti di governance.
3. Dopo aver generato un elenco, dai la priorità alle voci relative a migliorie e inizia a implementarle nel tuo carico di lavoro.
    a. Utilizza servizi come AWS Configcreare governance-as-code e convalidare il rispetto dei requisiti di governance.
    b. Utilizzando AWS Organizations, puoi avvalerti di policy di controllo dei servizi per l'implementazione dei requisiti di governance.
4. Fornisci la documentazione che convalida l'implementazione.

Livello di impegno per il piano di implementazione: medio L'implementazione di requisiti di governance mancanti può causare la rielaborazione del tuo carico di lavoro.

Risorse

Best practice correlate:

- OPS01-BP04 Valutazione dei requisiti di conformità: la conformità è come la governance, ma è esterna rispetto all'organizzazione.

Documenti correlati:

- AWS Guida all'ambiente cloud di gestione e governance
- Le migliori pratiche per le politiche di controllo dei AWS Organizations servizi in un ambiente con più account
- Governance in Cloud AWS: Il giusto equilibrio tra agilità e sicurezza
- Cosa sono la governance, il rischio e la conformità (GRC)?

Video correlati:

- AWS Gestione e governance: configurazione, conformità e audit - AWS Online Tech Talks
- AWS RE:InForce 2019: governance per l'era del cloud (-R1) DEM12
- AWS re:Invent 2020: raggiungi la conformità come codice utilizzando il codice AWS Config
- AWS re:Invent 2020: governance agile su AWS GovCloud (US)

Esempi correlati:

- AWS Config Esempi di Conformance Pack

Servizi correlati:

- AWS Config
- AWS Organizations - Politiche di controllo dei servizi

OPS01-BP04 Valutazione dei requisiti di conformità

I requisiti di conformità interna, di settore e normativa sono un fattore importante per la definizione delle priorità della tua organizzazione. L'assetto di conformità della tua azienda potrebbe impedirti di usare tecnologie specifiche o posizioni geografiche. Applica la due diligence in assenza di contesti di conformità esterni. Genera audit o report per convalidare la conformità.

Se comunichi all'esterno che il tuo prodotto è in linea con standard specifici di conformità, devi disporre di un processo interno in grado di garantire in modo costante la conformità. Gli esempi di standard di conformità includono PCI DSS, FedRAMP e HIPAA. Gli standard di conformità applicabili vengono stabiliti in base a diversi fattori, come il tipo di dati che la soluzione archivia o trasmette e quali aree geografiche sono supportate dalla soluzione.

Risultato desiderato:

- Requisiti di conformità interni, di settore e normativi sono integrati nella selezione dell'architettura.
- Puoi verificare la conformità e generare report di audit.

Anti-pattern comuni:

- Parti del tuo carico di lavoro rientrano nel framework Payment Card Industry Data Security Standard (PCI-DSS), ma il tuo carico di lavoro archivia dati di carte di credito non crittografati.
- Architetti e sviluppatori software non conoscono il contesto di conformità che la tua organizzazione è tenuta a rispettare.
- L'audit annuale Systems and Organizations Control (SOC2) Type II avrà luogo a breve e tu non sei in grado di verificare la presenza dei controlli richiesti.

Vantaggi dell'adozione di questa best practice:

- Grazie alla valutazione e comprensione dei requisiti di conformità applicati al carico di lavoro, sarà possibile organizzare le attività in base a priorità e offrire valore aggiunto.
- Scegli le sedi e le tecnologie corrette, in linea con il tuo contesto di integrità.
- La progettazione del tuo carico di lavoro ai fini degli audit ti consente di dimostrare il rispetto del modello di conformità.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

L'implementazione di questa best practice significa integrare requisiti di conformità nel processo di progettazione dell'architettura. I membri del tuo team sono a conoscenza del contesto di conformità richiesto. Convalidi la conformità in linea con il contesto.

Esempio del cliente

AnyCompany Retail archivia informazioni sulle carte di credito per i clienti. Gli sviluppatori del team di archiviazione delle carte sono al corrente della necessità di rispettare la conformità agli standard PCI-DSS. Hanno adottato misure per verificare che le informazioni sulle carte di credito siano archiviate e consultabili in totale sicurezza, in linea con quanto stabilito dagli standard PCI-DSS: Ogni anno collaborano con il team di sicurezza per confermare la conformità.

Passaggi dell'implementazione

1. Collabora con i team di sicurezza e governance per stabilire le conformità interne, normative o di settore deve rispettare il tuo carico di lavoro. Integra gli standard di conformità nel tuo carico di lavoro.
   a. Convalida la conformità continua delle risorse AWS con servizi come AWS Compute Optimizer e AWS Security Hub CSPM.

2. Comunica ai membri del tuo team i requisiti di conformità, in modo che possano gestire e far evolvere il carico di lavoro in linea con essi. I requisiti di conformità devono essere inclusi nelle scelte tecnologiche e architetturali.
3. A seconda del contesto di conformità, potresti dover generare un report di audit o conformità. Collabora con la tua organizzazione per automatizzare il più possibile questo processo.
   a. Utilizza servizi come AWS Audit Manager per convalidare la conformità e generare report di audit.
   b. Puoi scaricare documenti di sicurezza e conformità di AWS con AWS Artifact.

Livello di impegno per il piano di implementazione: medio Implementare i requisiti di conformità può essere complesso. Generare report di audit o documenti di conformità aggiunge altre complessità.

Risorse

Best practice correlate:

- SEC01-BP03 Identificazione e convalida degli obiettivi di controllo: gli obiettivi di controllo della sicurezza sono una parte importante della conformità generale.
- SEC01-BP06 Automatizzazione dei test e della convalida dei controlli di sicurezza nelle pipeline: nell'ambito delle tue pipeline, convalida i controlli di sicurezza. Puoi anche generare la documentazione di conformità per le nuove modifiche.
- SEC07-BP02 Applicazione di controlli di protezione dei dati in base alla loro sensibilità: molti framework di conformità si basano su policy di gestione e archiviazione dei dati.
- SEC10-BP03 Preparazione di funzionalità forensi: in alcuni casi, le funzionalità forensi consentono di verificare la conformità.

Documenti correlati:

- Centro AWS per la conformità
- Risorse per la conformità AWS
- Whitepaper su rischio e conformità AWS
- Modello di responsabilità condivisa AWS
- Servizi AWS coperti dal programma di conformità

Video correlati:

- AWS re:Invent 2020: Achieve compliance as code using AWS Compute Optimizer
- AWS re:Invent 2021 - Cloud compliance, assurance, and auditing
- AWS Summit ATL 2022 - Implementing compliance, assurance, and auditing on AWS (COP202)

Esempi correlati:

- PCI DSS and AWS Foundational Security Best Practices on AWS

Servizi correlati:

- AWS Artifact
- AWS Audit Manager
- AWS Compute Optimizer
- AWS Security Hub CSPM

OPS01-BP05 Valuta il panorama delle minacce

Valuta le minacce per l'azienda (ad esempio, concorrenza, rischi e responsabilità aziendali, rischi operativi e minacce per la sicurezza delle informazioni) e conserva le informazioni aggiornate in un registro dei rischi. Quando stabilisci dove concentrare gli sforzi, tieni in considerazione l'impatto dei rischi.

Il Framework Well-Architected enfatizza formazione, misurazione e miglioramento. Fornisce un approccio coerente per valutare le architetture e implementare progetti scalabili nel tempo. AWS fornisce l'assistenza necessaria AWS Well-Architected Toolper rivedere l'approccio prima dello sviluppo, lo stato dei carichi di lavoro prima della produzione e lo stato dei carichi di lavoro in produzione. Puoi confrontarli con le migliori pratiche AWS architettoniche più recenti, monitorare lo stato generale dei carichi di lavoro e ottenere informazioni sui potenziali rischi.

AWS i clienti hanno diritto a una revisione guidata Well-Architected dei loro carichi di lavoro mission-critical per misurare le loro architetture rispetto alle migliori pratiche. AWS I clienti del supporto Enterprise possono usufruire di una revisione delle operazioni, ideata per agevolare l'identificazione di lacune nell'approccio da loro utilizzato nel cloud.

Il coinvolgimento trasversale dei team per tali controlli aiuta a comprendere a livello comune i carichi di lavoro e il contributo dei ruoli del team al successo. Le esigenze identificate nel corso dell'analisi possono aiutarti a definire le priorità.

AWS Trusted Advisor è uno strumento che fornisce l'accesso a una serie di controlli di base che propongono ottimizzazioni utili per la definizione delle tue priorità. I clienti del supporto Business ed Enterprise hanno accesso a ulteriori controlli a livello di sicurezza, affidabilità, prestazioni e ottimizzazione dei costi, utili per definire le loro priorità.

Risultato desiderato:

- Esamini e agisci regolarmente in base a Well-Architected Trusted Advisor e ai risultati
- Sei a conoscenza dello stato delle patch più recenti dei servizi.
- Comprendi il rischio e l'impatto delle minacce note e intervieni di conseguenza.
- Implementi le mitigazioni necessarie.
- Comunichi azioni e contesto.

Anti-pattern comuni:

- Utilizzo della versione precedente di una libreria software nel tuo prodotto. Mancata conoscenza di aggiornamenti di sicurezza alla libreria per problemi che potrebbero avere un impatto imprevisto sul carico di lavoro.
- Rilascio da parte di un tuo concorrente di una versione del proprio prodotto che risolve i reclami di molti dei tuoi clienti relativi al tuo prodotto. Non hai dato priorità alla risoluzione di questi problemi noti.
- Perseguimento da parte delle autorità di regolamentazione di aziende come la tua, non conformi ai requisiti di conformità alla normativa legale. Mancata assegnazione della priorità ai requisiti di conformità in sospeso.

Vantaggi dell'adozione di questa best practice: identifichi e comprendi le minacce per la tua organizzazione e il tuo carico di lavoro ti consentono di determinare quali minacce affrontare, la loro priorità e le risorse necessarie per farlo.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

- Valuta il panorama delle minacce: valute le minacce per l'azienda (ad esempio, concorrenza, rischi e responsabilità aziendali, rischi operativi e minacce alla sicurezza delle informazioni), in modo da poterne includere l'impatto nel determinare dove concentrare le attività.

  - Bollettini sulla sicurezza AWS aggiornati
  - AWS Trusted Advisor
- Mantieni un modello delle minacce: definisci e mantieni un modello delle minacce che identifichi potenziali minacce, mitigazioni pianificate e predisposte e la relativa priorità. Esamina la probabilità che le minacce si manifestino come incidenti, il costo del ripristino dagli incidenti, il danno previsto causato e il costo per prevenire tali incidenti. Modifica le priorità man mano che i contenuti del modello di minaccia cambiano.

Risorse

Best practice correlate:

- SEC01-BP07 Identifica le minacce e dai priorità alle mitigazioni utilizzando un modello di minaccia

Documenti correlati:

- Conformità di Cloud AWS
- Bollettini sulla sicurezza AWS aggiornati
- AWS Trusted Advisor

Video correlati:

- AWS re:Inforce 2023 - A tool to help improve your threat modeling

OPS01-BP06 Valutazione dei compromessi gestendo vantaggi e rischi

Gli interessi contrastanti di più parti possono complicare l'assegnazione delle priorità a impegni, sviluppo delle capacità e conseguimento di risultati in linea con le strategie aziendali. Ad esempio, è possibile che ti venga chiesto di accelerare la commercializzazione di nuove funzionalità anziché ottimizzare i costi dell'infrastruttura IT. Questa richiesta può mettere due parti interessate in conflitto reciproco. In queste situazioni, le decisioni devono essere prese da un'autorità superiore che risolve il conflitto. I dati sono necessari per rimuovere l'aspetto emotivo dal processo decisionale.

La stessa sfida può verificarsi a livello strategico. Ad esempio, la scelta tra l'utilizzo di tecnologie di database relazionali o non relazionali può avere un impatto significativo sul funzionamento di un'applicazione. È fondamentale comprendere i risultati prevedibili delle varie scelte.

AWS può aiutarti a sensibilizzare i team su AWS e i suoi servizi, affinché comprendano meglio l'impatto delle loro scelte sul carico di lavoro. Per istruire i tuoi team, utilizza le risorse fornite da Supporto (Centro conoscenze AWS, AWS Discussion Forums e Supporto Center) e la documentazione AWS. Per ulteriori domande, contatta Supporto.

AWS condivide inoltre le best practice e i modelli appresi attraverso la gestione nella Amazon Builders' Library. Un'ampia gamma di ulteriori informazioni utili è disponibile tramite il blog AWS e il podcast ufficiale di AWS.

Risultato desiderato: presenza di un framework di governance decisionale definito in modo chiaro per semplificare le decisioni importanti a tutti i livelli all'interno dell'organizzazione di distribuzione del cloud. Questo framework include funzionalità come registro dei rischi, ruoli definiti autorizzati a prendere decisioni e modelli prestabiliti per ogni livello di decisione adottabile. Il framework definisce in anticipo le modalità di risoluzione dei conflitti, quali dati vanno presentati e come viene stabilita la priorità delle opzioni, in modo che una volta prese le decisioni sia subito possibile lavorare per applicarle. Il framework del processo decisionale include un approccio standardizzato alla revisione e alla valutazione di vantaggi e rischi di ogni decisione per comprenderne i compromessi. Ciò può comprendere fattori esterni, come l'aderenza ai requisiti di conformità normativa.

Anti-pattern comuni:

- Richiesta degli investitori di dimostrare la conformità agli standard PCI DSS (Payment Card Industry Data Security Standard). Non prendi in considerazione i compromessi tra soddisfare la loro richiesta e continuare con le attività di sviluppo già in corso. Al contrario, prosegui con il lavoro di sviluppo senza dimostrare la conformità. Gli investitori interrompono il supporto all'azienda a causa dei dubbi relativi alla sicurezza della piattaforma e dei loro investimenti.
- Si è deciso di includere una libreria che uno sviluppatore ha trovato su Internet. Non hai valutato i rischi derivanti dall'adozione di questa libreria da un'origine sconosciuta e non sai se contiene vulnerabilità o codice dannoso.
- Giustificazione aziendale originale per la migrazione basata sulla modernizzazione del 60% dei carichi di lavoro delle applicazioni. Tuttavia, a causa di difficoltà tecniche, è stata presa la decisione di modernizzare solo il 20%, con una riduzione dei vantaggi pianificati a lungo termine, un maggiore impegno operativo dei team dell'infrastruttura per supportare manualmente i sistemi legacy e un accresciuto affidamento sullo sviluppo di nuove competenze nei team dell'infrastruttura che non avevano pianificato questo cambiamento.

Vantaggi dell'adozione di questa best practice: allineamento e supporto completi delle priorità aziendali a livello gestionale, comprensione dei rischi legati al raggiungimento del successo, decisioni informate e azioni opportune quando i rischi costituiscono un ostacolo per le possibilità di successo. Comprendere implicazioni e conseguenze delle tue decisioni ti aiuta a stabilire le priorità delle opzioni, oltre a ottenere l'accordo dei leader più rapidamente, fornendo risultati aziendali migliori. L'identificazione dei benefici disponibili delle tue scelte e la consapevolezza dei rischi per la tua organizzazione ti aiutano a prendere decisioni basate sui dati, piuttosto che affidarti agli aneddoti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

La gestione di vantaggi e rischi va definita da un organo direttivo che stabilisca i requisiti del processo decisionale chiave. Le decisioni devono essere prese e la priorità va assegnata in base ai vantaggi derivanti per l'organizzazione, con una comprensione dei rischi connessi. L'accuratezza delle informazioni è fondamentale quando si prendono le decisioni organizzative, che devono basarsi su misurazioni affidabili ed essere definite secondo le procedure comuni del settore per l'analisi costi-benefici. Per prendere questo tipo di decisioni, occorre trovare un equilibrio tra l'autorità centralizzata e quella decentralizzata. Esiste sempre un compromesso ed è importante capire l'impatto di ogni scelta sulle strategie definite e sui risultati aziendali desiderati.

Passaggi dell'implementazione

1. Formalizza le procedure di misurazione dei vantaggi in un framework olistico di governance del cloud.
   a. Bilancia il controllo centrale del processo decisionale con l'autorità decentralizzata per alcune decisioni.
   b. Riconosci che i gravosi processi decisionali imposti per ogni decisione possono rallentare le operazioni.
   c. Incorpora nel processo decisionale fattori esterni, come i requisiti di conformità.
2. Stabilisci un framework del processo decisionale concordato per vari livelli di decisioni, che includa chi è tenuto a prendere le decisioni soggette a conflitti di interessi.
   a. Centralizza le decisioni definitive che potrebbero essere irreversibili.
   b. Consenti ai leader dell'organizzazione di livello inferiore di prendere decisioni reversibili.
3. Comprendi e gestisci i vantaggi e i rischi. Bilancia i vantaggi delle decisioni rispetto ai rischi connessi.

a. Identificazione dei vantaggi: identifica i vantaggi in base a obiettivi aziendali, esigenze e priorità, ad esempio l'impatto del caso aziendale, il time-to-market, la sicurezza, l'affidabilità, le prestazioni e i costi.

b. Identificazione dei rischi: identifica i rischi in base a obiettivi aziendali, esigenze e priorità, ad esempio il time-to-market, la sicurezza, l'affidabilità, le prestazioni e il costo.

c. Valutazione dei vantaggi rispetto ai rischi e decisioni informate: determina l'impatto di vantaggi e rischi in base a obiettivi, esigenze e priorità delle principali parti interessate, inclusi business, sviluppo e operazioni. Valuta il valore del vantaggio rispetto alla probabilità di concretizzazione del rischio e al costo del suo impatto. Ad esempio, enfatizzare la velocità di accesso al mercato rispetto all'affidabilità potrebbe offrire un vantaggio competitivo. Tuttavia, ciò potrebbe causare tempi di attività ridotti in presenza di problemi di affidabilità.

4. Applica in modo programmatico le decisioni chiave che automatizzano l'aderenza ai requisiti di conformità.
5. Impiega funzionalità e framework noti del settore, come Value Stream Analysis e LEAN, per definire la linea di base per prestazioni dello stato attuale, metriche aziendali e iterazioni dei progressi verso il miglioramento di tali metriche.

Livello di impegno per il piano di implementazione: medio-alto

Risorse

Best practice correlate:

- OPS01-BP05 Valutazione del panorama delle minacce

Documenti correlati:

- Elementi della cultura del Giorno 1 di Amazon | Adotta decisioni di alta qualità e ad alta velocità
- Governance del cloud
- Management & Governance Cloud Environment
- Governance in the Cloud and in the Digital Age: Parts One & Two

Video correlati:

- Podcast | Jeff Bezos | On how to make decisions

Esempi correlati:

- Make informed decisions using data (The DevOps Sagas)
- Using development value stream mapping to identify constraints to DevOps outcomes

OPS 2. Come strutturare la tua organizzazione per supportare i risultati aziendali?

I tuoi team devono comprendere quale contributo offrono nel raggiungimento dei risultati aziendali. I team devono avere obiettivi condivisi e comprendere il proprio ruolo nel successo degli altri team. Comprendere la responsabilità, la proprietà, il modo in cui vengono prese le decisioni e chi ha l'autorità decisionale aiuterà a concentrare gli sforzi e a ottimizzare i contributi dei team.

Best practice

- OPS02-BP01 Le risorse hanno identificato i proprietari
- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure
- OPS02-BP03 Assegnazione di proprietari identificati alle operazioni che siano responsabili delle relative prestazioni
- OPS02-BP04 Definizione di meccanismi per gestire responsabilità e titolarità
- OPS02-BP05 Definizione di meccanismi per richiedere aggiunte, modifiche ed eccezioni
- OPS02-BP06 Predefinizione o negoziazione delle responsabilità tra i team

OPS02-BP01 Le risorse hanno identificato i proprietari

Le risorse per il tuo carico di lavoro devono disporre di proprietari identificati per il controllo delle modifiche, la risoluzione dei problemi e altre funzioni. I proprietari sono assegnati a carichi di lavoro, account, infrastrutture, piattaforme e applicazioni. La registrazione della proprietà avviene tramite strumenti come un registro centrale o metadati collegati alle risorse. Il valore aziendale dei componenti è alla base dei processi e delle procedure applicate.

Risultato desiderato:

- Le risorse presentano proprietari identificati tramite i metadati o un registro centrale.
- I membri del team possono identificare chi è il proprietario delle risorse.
- Gli account hanno un solo proprietario, laddove possibile.

Anti-pattern comuni:

- I tuoi contatti alternativi non sono popolati. Account AWS
- Risorse prive di tag che identificano i team proprietari.
- Hai una ITSM coda senza una mappatura delle email.
- Due team con entrambi la proprietà di una parte critica dell'infrastruttura.

Vantaggi dell'adozione di questa best practice:

- Il controllo delle modifiche per le risorse è immediato con la proprietà assegnata.
- Puoi coinvolgere i proprietari corretti quando risolvi i problemi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Definisci qual è il significato della proprietà per i casi d'uso delle risorse nel tuo ambiente. Proprietà significa supervisionare le modifiche alla risorsa, supportare la risorsa durante la risoluzione dei problemi o essere finanziariamente affidabile. Specifica e registra i proprietari delle risorse, con nome, informazioni di contatto, organizzazione e team.

Esempio del cliente

AnyCompany La vendita al dettaglio definisce la proprietà come il team o l'individuo responsabile delle modifiche e del supporto alle risorse. Sfruttano AWS Organizations per gestire le proprie Account AWS. Contatti alternativi degli account sono configurati con caselle di posta di gruppo. Ogni ITSM coda è associata a un alias e-mail. I tag identificano chi possiede le risorse. AWS Per altre piattaforme e infrastrutture, è presente una pagina wiki che identifica proprietà e informazioni di contatto.

Passaggi dell'implementazione

1. Inizia definendo la proprietà dell'organizzazione. La proprietà può significare essere proprietari del rischio collegato alla risorsa, delle modifiche alla risorsa o supportare la stessa durante la risoluzione dei problemi. Proprietà può anche significare proprietà amministrativa o finanziaria della risorsa.
2. Usa AWS Organizations per gestire gli account. Puoi gestire a livello centrale i contatti alternativi per gli account.
   a. Se usi indirizzi e-mail e numeri di telefono aziendali come informazioni di contatto, puoi accedervi anche se le persone a cui appartengono non fanno più parte dell'organizzazione.

Ad esempio, crea elenchi di distribuzione delle e-mail separati per fatturazione, operazioni e sicurezza e configurali come contatti per Fatturazione, Sicurezza e Operazioni in ogni Account AWS attivo. Più persone riceveranno AWS notifiche e saranno in grado di rispondere, anche se qualcuno è in vacanza, cambia ruolo o lascia l'azienda.

b. Se un account non è gestito da AWS Organizations, i contatti alternativi dell'account aiutano AWS a contattare il personale opportuno, se necessario. Configura i contatti alternativi dell'account per indirizzare le persone a un gruppo invece che a un individuo.

3. Utilizza i tag per identificare i proprietari AWS delle risorse. Puoi specificare i proprietari e le loro informazioni di contatto in tag separati.
   a. Puoi utilizzare le regole di AWS Config per far sì che le risorse presentino i tag di proprietà richiesti.
   b. Per una guida approfondita su come creare una strategia di tagging per la tua organizzazione, consulta il whitepaper AWS Tagging Best Practices.
4. Usa Amazon Q Business, un assistente conversazionale che utilizza l'IA generativa per migliorare la produttività della forza lavoro, rispondere a domande e completare attività in base alle informazioni presenti nei sistemi aziendali.
   a. Collega Amazon Q Business all'origine dati della tua azienda. Amazon Q Business offre connettori predefiniti per oltre 40 fonti di dati supportate, tra cui Amazon Simple Storage Service (Amazon S3), SharePoint Microsoft, Salesforce e Atlassian Confluence. Per ulteriori informazioni, consulta Connettori di Amazon Q Business.
5. Per altre risorse, piattaforme e infrastrutture, crea la documentazione che stabilisce la proprietà. Tutti i membri del team devono poter accedere a queste informazioni.

Livello di impegno per il piano di implementazione: basso Sfrutta le informazioni di contatto e i tag dell'account per assegnare la proprietà delle risorse. AWS Per altre risorse puoi usare qualcosa di semplice come una tabella in un wiki per registrare la proprietà e le informazioni di contatto, oppure usare ITSM uno strumento per mappare la proprietà.

Risorse

Best practice correlate:

- OPS02-BP02 I processi e le procedure hanno identificato i proprietari
- OPS02-BP04 Esistono meccanismi per gestire le responsabilità e la proprietà

Documenti correlati:

- AWS Account Management - Updating contact information
- AWS Organizations - Aggiornamento dei contatti alternativi all'interno dell'organizzazione
- Whitepaper AWS Tagging Best Practices
- Crea app di intelligenza artificiale generativa aziendali private e sicure con Amazon Q Business e AWS IAM Identity Center
- Amazon Q Business, now generally available, helps boost workforce productivity with generative AI
- Cloud AWS Blog Operations & Migrations - Implementazione di controlli di tagging automatizzati e centralizzati con e AWS ConfigAWS Organizations
- AWS Blog sulla sicurezza - Estendi i tuoi hook di pre-commit con AWS CloudFormation Guard
- AWS DevOps Blog - Integrazione AWS CloudFormation Guard nelle pipeline CI/CD

Workshop correlati:

- Workshop AWS - Tagging

Esempi correlati:

- Regole di AWS Config - Amazon EC2 con tag obbligatori e valori validi

Servizi correlati:

- Regole di AWS Config - tag obbligatori
- AWS Organizations

OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure

È utile sapere chi ha la proprietà della definizione di singoli processi e procedure, poiché tali processi e procedure specifici vengono utilizzati e perché tale proprietà esiste. Comprendere i motivi per cui vengono utilizzati processi e procedure specifici aiuta a identificare le opportunità di miglioramento.

Risultato desiderato: la tua organizzazione dispone di una serie di processi e procedure per le attività operative ben definiti e gestiti. L'archiviazione di processi e procedure avviene in una posizione centrale e questi sono a disposizione dei membri del team. I processi e le procedure vengono

aggiornati di frequente attraverso l'assegnazione chiara della proprietà. Ove possibile, script, modelli e documenti di automazione vengono implementati come codice.

Anti-pattern comuni:

- Mancata documentazione dei processi. È possibile la presenza di script frammentati su workstation degli operatori isolate.
- Conoscenza relativa all'uso degli script nelle mani di pochi individui oppure l'acquisizione avviene in modo informale come conoscenza di team.
- Necessità di aggiornare un processo legacy, ma manca chiarezza circa la proprietà dell'aggiornamento e l'autore originale non fa più parte dell'organizzazione.
- Non è possibile individuare processi e script, quindi non sono immediatamente disponibili quando necessario (ad esempio, in risposta a un incidente).

Vantaggi dell'adozione di questa best practice:

- Processi e procedure incentivano l'impegno nella gestione dei carichi di lavoro.
- I nuovi membri del team diventano efficienti in modo più rapido.
- Riduzione dei tempi di mitigazione degli incidenti.
- Membri del team (e team) diversi possono utilizzare gli stessi processi e procedure in modo coerente.
- I team procedono a scalare i processi tramite processi ripetibili.
- Processi e procedure standardizzati aiutano a mitigare l'impatto del trasferimento delle responsabilità del carico di lavoro tra i team.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

- Esistono proprietari identificati di processi e procedure, responsabili della loro definizione.
  - Identifica le attività operative eseguite a supporto dei carichi di lavoro. Documenta queste attività in un percorso individuabile.
  - Identifica in modo univoco la persona o il team responsabile della specifica di un'attività. Questo soggetto deve verificare la possibilità che questa possa essere correttamente eseguita dal componente di un team con opportune competenze, dotato di autorizzazioni, accesso e

strumenti adeguati. In caso di problemi nello svolgimento di tale attività, i membri del team che la eseguono sono responsabili della redazione di feedback dettagliati necessari per migliorarla.

- Acquisisci la responsabilità dei metadati dell'artefatto dell'attività tramite servizi come AWS Systems Manager, documenti e AWS Lambda. Acquisisci la responsabilità delle risorse utilizzando tag o gruppi di risorse, specificando proprietà e informazioni di contatto. Utilizza AWS Organizations per creare policy di tagging e garantire l'acquisizione di proprietà e informazioni di contatto.

- Nel tempo, queste procedure si evolvono per essere eseguibili come codice, riducendo la necessità dell'intervento umano.
  - Ad esempio, prendi in considerazione le funzioni AWS Lambda, i modelli CloudFormation o i documenti di automazione di AWS Systems Manager.
  - Esegui il controllo delle versioni nei repository appropriati.
  - Applica i tag adeguati alle risorse, in modo da agevolare l'identificazione di proprietari e documentazione.

Esempio del cliente

AnyCompany Retail definisce come proprietario il team o l'individuo responsabile dei processi per un'applicazione o gruppi di applicazioni (che condividono procedure e tecnologie architetturali comuni). Inizialmente, processi e procedure vengono documentati nel sistema di gestione dei documenti come guide dettagliate, individuabili tramite i tag dell'Account AWS che ospita l'applicazione e di gruppi specifici di risorse all'interno dell'account. AnyCompany Retail si avvale di AWS Organizations per gestire gli Account AWS. Nel tempo, questi processi vengono convertiti in codice e le risorse vengono definite utilizzando l'infrastructure as code (ad esempio CloudFormation o modelli AWS Cloud Development Kit (AWS CDK)). I processi operativi diventano documenti di automazione in AWS Systems Manager o funzioni di AWS Lambda, avviabili come attività pianificate in risposta a eventi, ad esempio allarmi AWS di CloudWatch o eventi AWS di EventBridge, oppure avviati da richieste all'interno di una piattaforma di gestione dei servizi IT (ITSM). Tutti i processi dispongono di tag per l'identificazione della proprietà. La documentazione per l'automazione e il processo viene mantenuta all'interno delle pagine wiki generate dal repository di codice per il processo.

Passaggi dell'implementazione

1. Documenta processi e procedure esistenti.
   a. Rivedili e mantienili aggiornati.

b. Identifica un proprietario per ciascun processo o procedura.

c. Applica a ognuno il controllo delle versioni.

d. Ove possibile, condividi processi e procedure tra carichi di lavoro e ambienti che condividono progetti architetturali.

2. Stabilisci meccanismi di feedback e miglioramento.

a. Definisci policy relative alla frequenza di revisione dei processi.

b. Definisci i processi per revisori e approvatori.

c. Implementa i problemi o crea una coda di ticket per fornire e monitorare il feedback.

d. Ove possibile, i processi e le procedure vanno approvati preventivamente e classificati in base ai rischi da parte di un comitato di approvazione delle modifiche (CAB).

3. Verifica che processi e procedure siano accessibili e individuabili da chi deve eseguirli.

a. Utilizza i tag per indicare dove è possibile accedere a processi e procedure per il carico di lavoro.

b. Utilizza messaggi di errore ed eventi significativi per indicare processi o procedure appropriati per risolvere un problema.

c. Usa i wiki e la gestione dei documenti per rendere processi e procedure consultabili in modo coerente in tutta l'organizzazione.

4. Usa Amazon Q Business, un assistente conversazionale che utilizza l'IA generativa per migliorare la produttività della forza lavoro, rispondere a domande e completare attività in base alle informazioni presenti nei sistemi aziendali.

a. Collega Amazon Q Business all'origine dati della tua azienda. Amazon Q Business offre connettori predefiniti per oltre 40 origini dati supportate, tra cui Amazon S3, Microsoft SharePoint, Salesforce e Atlassian Confluence. Per ulteriori informazioni, consulta Connettori di Amazon Q.

5. Automatizza quando appropriato.

a. È opportuno eseguire le automazioni quando servizi e tecnologie forniscono un'API.

b. Fornisci indicazioni adeguate in merito ai processi. Sviluppa casi utente e requisiti per automatizzare i processi.

c. Misura correttamente l'uso di processi e procedure e crea problemi o ticket come un'opportunità di miglioramento continuo.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS02-BP01 Associazione di proprietari identificati alle risorse
- OPS02-BP04 Definizione di meccanismi per gestire responsabilità e titolarità
- OPS11-BP04 Gestione delle informazioni

Documenti correlati:

- AWS Whitepaper - Introduzione a DevOps in AWS
- Whitepaper AWS - Best Practices for Tagging AWS Resources
- Whitepaper AWS: Organizing Your AWS Environment Using Multiple Accounts
- Cloud AWS Operations and Migrations Blog - Using Amazon Q Business to streamline your operations
- Cloud AWS Post del blog Operations & Migrations - Build a Cloud Automation Practice for Operational Excellence: Best Practices from AWS Managed Services
- Post del blog Cloud AWS Operations & Migrations - Implementing automated and centralized tagging controls with AWS Config and AWS Organizations
- AWS Post del blog Security - Extend your pre-commit hooks with AWS CloudFormation Guard
- Post del blog AWS DevOps - Integrating AWS CloudFormation Guard into CI/CD pipelines

Workshop correlati:

- AWS Well-Architected Operational Excellence Workshop
- AWS Workshop - Tagging

Video correlati:

- How to automate IT Operations on AWS
- AWS re:Invent 2020 - Automate anything with AWS Systems Manager
- AWS re:Inforce 2022 - Automating patch management and compliance using AWS (NIS306)
- Supportos You - Diving Deep into AWS Systems Manager

Servizi correlati:

- AWS Systems Manager - Automation
- AWS Service Management Connector

OPS02-BP03 Assegnazione di proprietari identificati alle operazioni che siano responsabili delle relative prestazioni

È utile sapere chi ha la responsabilità di eseguire attività specifiche su carichi di lavoro definiti e perché tale responsabilità esiste. Conoscere chi ha la responsabilità di eseguire le attività fornisce indicazioni su chi eseguirà l'attività, chi convaliderà il risultato e chi fornirà feedback al proprietario dell'attività.

Risultato desiderato:

L'organizzazione definisce chiaramente le responsabilità per eseguire attività specifiche su carichi di lavoro stabiliti e rispondere agli eventi generati dai carichi di lavoro. L'organizzazione documenta la responsabilità dei processi e degli adempimenti e rende queste informazioni individuabili. Esamini e aggiorni le responsabilità in caso di cambiamenti organizzativi e i team monitorano e misurano le prestazioni delle attività di identificazione di difetti e inefficienze. Implementi i meccanismi di feedback per monitorare difetti e miglioramenti e supportare il miglioramento continuo.

Anti-pattern comuni:

- Mancata documentazione delle responsabilità.
- Esistono script frammentati su workstation degli operatori isolate. Solo poche persone sanno come usarli o li chiamano informalmente conoscenze del team.
- Necessità di aggiornare un processo legacy, ma non si sa chi è il proprietario e l'autore originale non fa più parte dell'organizzazione.
- Mancata possibilità di individuare processi e script, quindi non sono immediatamente disponibili quando necessario, ad esempio, in risposta a un incidente.

Vantaggi dell'adozione di questa best practice:

- Sai chi è responsabile dell'esecuzione di un'attività, a chi notificare un'azione necessaria e chi esegue l'azione, convalida il risultato e fornisce il feedback al titolare dell'attività.
- Processi e procedure incentivano l'impegno nella gestione dei carichi di lavoro.

- I nuovi membri del team diventano efficienti in modo più rapido.
- Riduci il tempo necessario per mitigare gli incidenti.
- Team diversi utilizzano medesimi processi e procedure per eseguire le attività in modo coerente.
- I team procedono a scalare i processi tramite processi ripetibili.
- Processi e procedure standardizzati aiutano a mitigare l'impatto del trasferimento delle responsabilità del carico di lavoro tra i team.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Per definire le responsabilità, inizia usando la documentazione esistente, ad esempio matrici di responsabilità, processi e procedure, ruoli e responsabilità, strumenti e automazione. Esamina la documentazione e organizza discussioni sulle responsabilità dei processi documentati. Collaborando con i team, identifica i disallineamenti tra le responsabilità e i processi documentati. Parla dei servizi offerti con i clienti interni dei team per identificare le divergenze nelle aspettative tra i team.

Analizza e risolvi le discrepanze. Identifica le opportunità di miglioramento e le attività richieste di frequente e con uso intensivo di risorse, in genere ottime candidate al miglioramento. Esamina best practice, modelli e linee guida prescrittive per semplificare e standardizzare i miglioramenti. Registra le opportunità di miglioramento e monitora i miglioramenti fino al completamento.

Nel tempo, queste procedure si evolvono per essere eseguibili come codice, riducendo la necessità dell'intervento umano. Ad esempio, è possibile avviare le procedure come funzioni AWS Lambda, modelli CloudFormation o documenti di automazione AWS Systems Manager. Verifica che queste procedure siano sottoposte al controllo delle versioni nei repository appropriati e includano i corretti tag delle risorse in modo che i team possano identificare prontamente responsabili e documentazione. Documenta la responsabilità dello svolgimento delle attività, quindi monitora l'avvio e il funzionamento delle automazioni, nonché le prestazioni dei risultati desiderati.

Esempio del cliente

AnyCompany Retail definisce come proprietario il team o l'individuo responsabile dei processi per un'applicazione o gruppi di applicazioni (che condividono procedure e tecnologie architetturali comuni). Inizialmente, l'azienda documenta processi e procedure come guide dettagliate nel sistema di gestione dei documenti. Rende le procedure individuabili applicando i tag nell'Account AWS che ospita l'applicazione e in gruppi specifici di risorse dell'account, utilizzando AWS Organizations

per gestire gli Account AWS. Nel tempo, AnyCompany Retail converte questi processi in codice e definisce le risorse utilizzando l'infrastructure as code, tramite servizi come CloudFormation o modelli AWS Cloud Development Kit (AWS CDK). I processi operativi diventano documenti di automazione in AWS Systems Manager o funzioni di AWS Lambda, avviabili come attività pianificate in risposta a eventi, ad esempio allarmi di Amazon CloudWatch o eventi di Amazon EventBridge, oppure avviati da richieste all'interno di una piattaforma di gestione dei servizi IT (ITSM). Tutti i processi dispongono dei tag per identificare il proprietario. I team gestiscono la documentazione per l'automazione e il processo nelle pagine wiki generate dal repository di codice per il processo.

Passaggi dell'implementazione

1. Documenta processi e procedure esistenti.
    a. Esamina e verifica che siano aggiornati.
    b. Verifica che ogni processo o procedura abbia un proprietario.
    c. Applica alle procedure il controllo delle versioni.
    d. Ove possibile, condividi processi e procedure tra carichi di lavoro e ambienti che condividono progetti architetturali.
2. Stabilisci meccanismi di feedback e miglioramento.
    a. Definisci policy relative alla frequenza di revisione dei processi.
    b. Definisci i processi per revisori e approvatori.
    c. Implementa i problemi o crea una coda di ticket per fornire e monitorare il feedback.
    d. Ove possibile, i processi e le procedure devono essere approvati preventivamente e classificati in base ai rischi da parte di un comitato di approvazione delle modifiche (CAB).
3. Rendi i processi e le procedure accessibili e individuabili dagli utenti che devono eseguirli.
    a. Utilizza i tag per indicare dove è possibile accedere a processi e procedure per il carico di lavoro.
    b. Utilizza messaggi di errore ed eventi significativi per indicare il processo o la procedura appropriata per risolvere il problema.
    c. Usa i wiki o la gestione dei documenti per rendere i processi e le procedure consultabili in modo coerente in tutta l'organizzazione.
4. Automatizza quando è opportuno farlo.
    a. Laddove servizi e tecnologie forniscono un'API, sviluppa le automazioni.
    b. Verifica che i processi siano ben compresi e sviluppa casi utente e requisiti per automatizzare i processi.

c. Misura l'uso corretto di processi e procedure e sfrutta i problemi per supportare il miglioramento continuo.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS02-BP01 Associazione di proprietari identificati alle risorse
- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure
- OPS02-BP04 Definizione di meccanismi per gestire responsabilità e titolarità
- OPS02-BP05 Definizione di meccanismi per identificare responsabilità e proprietà
- OPS11-BP04 Gestione delle informazioni

Documenti correlati:

- Whitepaper AWS | Introduzione a DevOps in AWS
- Whitepaper AWS | Best Practices for Tagging AWS Resources
- Whitepaper AWS | Organizing Your AWS Environment Using Multiple Accounts
- Post del blog Cloud AWS Operations & Migrations | Build a Cloud Automation Practice for Operational Excellence: Best Practices from AWS Managed Services
- Workshop AWS - Tagging
- AWS Service Management Connector

Video correlati:

- AWS Knowledge Center Live | Tagging AWS Resources
- AWS re:Invent 2020 | Automate anything with AWS Systems Manager
- AWS re:Inforce 2022 | Automating patch management and compliance using AWS (NIS306)
- Supportos You | Diving Deep into AWS Systems Manager

OPS02-BP04 Definizione di meccanismi per gestire responsabilità e titolarità

Comprendi le responsabilità del tuo ruolo e il modo in cui contribuisci ai risultati aziendali in quanto questa conoscenza fornisce indicazioni sulle priorità delle tue attività e sul perché il tuo ruolo è importante. I membri del team possono quindi riconoscere le esigenze e rispondere in modo appropriato. Quando i membri del team comprendono il proprio ruolo, possono stabilire la titolarità, identificare le opportunità di miglioramento e capire come influenzare o apportare le modifiche appropriate.

Occasionalmente, una responsabilità potrebbe non avere un titolare definito. In queste situazioni, progetta un meccanismo per risolvere la lacuna. Crea un percorso di escalation ben definito a qualcuno con l'autorità di assegnare la responsabilità o il piano per risolvere il problema.

Risultato desiderato: responsabilità definite in modo chiaro per i team all'interno dell'organizzazione, che comprendono il modo in cui sono correlate alle risorse, alle azioni da eseguire, ai processi e alle procedure. Queste responsabilità sono in linea con le responsabilità e gli obiettivi del team, nonché con le responsabilità degli altri team. Documenti i percorsi di escalation in modo coerente e individuabile e inserisci queste decisioni in artefatti di documentazione, come matrici di responsabilità, definizioni di team o pagine wiki.

Anti-pattern comuni:

- Le responsabilità del team sono ambigue o mal definite.
- Il team non allinea i ruoli alle responsabilità.
- Il team non allinea scopi e obiettivi alle responsabilità, rendendo difficile misurare il successo delle attività.
- Le responsabilità dei membri del team non sono in linea con il team e l'organizzazione in generale.
- Il team non mantiene aggiornate le responsabilità rendendole incoerenti con le attività svolte dal team.
- I percorsi di escalation per determinare le responsabilità non sono definiti o non sono chiari.
- I percorsi di escalation non hanno un unico responsabile del thread per garantire una risposta tempestiva.
- Ruoli, responsabilità e percorsi di escalation non sono individuabili e quindi non sono immediatamente disponibili quando richiesto, ad esempio in risposta a un incidente.

Vantaggi dell'adozione di questa best practice:

- Una volta compreso chi ha la responsabilità o la titolarità, puoi contattare il team o il membro del team appropriato per effettuare una richiesta o trasferire un'attività.
- Per ridurre il rischio di inattività e di esigenze non soddisfatte, identifichi una persona che ha l'autorità di assegnare responsabilità o titolarità.
- Quando si definisce chiaramente l'ambito di una responsabilità, i membri del team acquisiscono autonomia e titolarità.
- Le tue responsabilità forniscono indicazioni sulle decisioni che prendi, sulle azioni che intraprendi e sulle tue attività di distribuzione ai titolari appropriati.
- Ti sarà facile identificare le responsabilità abbandonate perché hai una chiara comprensione di ciò che non rientra nelle responsabilità del tuo team e quindi potrai effettuare l'escalation per chiedere chiarimenti.
- I team evitano confusione e tensione e possono gestire in modo più adeguato i carichi di lavoro e le risorse.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Identifica i ruoli e le responsabilità dei membri del team e verifica che comprendano le aspettative del proprio ruolo. Rendi queste informazioni individuabili in modo che i membri della tua organizzazione possano identificare il team o la persona da contattare per esigenze specifiche. Man mano che le organizzazioni capitalizzano le opportunità di migrare e modernizzare su AWS, i ruoli e le responsabilità potrebbero cambiare. Rendi i team e i membri consapevoli delle loro responsabilità e offri la formazione appropriata per svolgere le attività durante questo cambiamento.

Determina il ruolo o il team che deve ricevere le escalation per identificare responsabilità e titolarità. Questo team può interagire con varie parti interessate per prendere le decisioni. Tuttavia, è proprietario della gestione del processo decisionale.

Fornisci ai membri della tua organizzazione meccanismi accessibili per scoprire e identificare titolarità e responsabilità. Questi meccanismi insegnano loro a chi rivolgersi per esigenze specifiche.

Esempio del cliente

AnyCompany Retail ha recentemente completato una migrazione dei carichi di lavoro da un ambiente on-premises alla zona di destinazione in AWS con un approccio lift and shift. Ha eseguito una revisione delle operazioni per esaminare come vengono svolte le attività operative comuni e ha verificato che la matrice di responsabilità esistente rifletta le operazioni nel nuovo ambiente. Quando

ha eseguito la migrazione dall'ambiente on-premises ad AWS, ha ridotto le responsabilità dei team dell'infrastruttura relative all'hardware e all'infrastruttura fisica. Questo passaggio ha anche rivelato nuove opportunità per evolvere il modello operativo dei carichi di lavoro.

Oltre ad aver identificato, risolto e documentato la maggior parte delle responsabilità, ha anche definito i percorsi di escalation per eventuali responsabilità mancanti o che potrebbero cambiare con l'evolversi delle procedure operative. Per la ricerca di nuove opportunità per standardizzare e migliorare l'efficienza dei carichi di lavoro, fornisci l'accesso a strumenti operativi come AWS Systems Manager e strumenti di sicurezza come AWS Security Hub CSPM e Amazon GuardDuty. AnyCompany Retail combina una revisione delle responsabilità e della strategia sulla base dei miglioramenti che intende eseguire per primi. Man mano che l'azienda adotta nuovi modi di lavorare e modelli tecnologici, aggiorna la propria matrice di responsabilità di conseguenza.

Passaggi dell'implementazione

1. Inizia con la documentazione esistente. Alcuni documenti di origine tipici possono essere:
   a. Matrici di responsabilità o responsabili, affidabili, consultabili e informate (RACI).
   b. Definizioni dei team o pagine wiki.
   c. Definizioni e offerte di servizi.
   d. Ruolo o descrizione delle mansioni lavorative.
2. Esamina la documentazione e organizza discussioni sulle responsabilità documentate:
   a. Collaborando con i team identifica i disallineamenti tra le responsabilità documentate e quelle normalmente assunte dai team.
   b. Esamina i potenziali servizi offerti dai clienti interni per identificare le lacune nelle aspettative tra i team.
3. Analizza e risolvi le discrepanze.
4. Identifica le opportunità di miglioramento.
   a. Identifica le richieste più frequenti e con uso intensivo di risorse, che in genere sono ottime candidate al miglioramento.
   b. Esamina le best practice, comprendi i modelli, segui le linee guida prescrittive per semplificare e standardizzare i miglioramenti.
   c. Registra le opportunità di miglioramento e monitorale fino al completamento.
5. Se nessuno nel team è responsabile della gestione e del monitoraggio dell'assegnazione delle responsabilità, identifica qualcuno che assuma tale responsabilità.
6. Definisci un processo per consentire ai team di richiedere chiarimenti sulla responsabilità.

a. Esamina il processo e verifica che sia chiaro e semplice da usare.
b. Assicurati che qualcuno sia proprietario e segua le escalation fino al completamento.
c. Stabilisci le metriche operative per misurare l'efficacia.
d. Crea un meccanismo di feedback per verificare che i team possano evidenziare le opportunità di miglioramento.
e. Implementa un meccanismo di revisione periodica.

7. Rendi i documenti disponibili in una posizione individuabile e accessibile.
   a. I wiki o il portale di documentazione sono le posizioni normalmente scelte.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS01-BP06 Valutazione dei compromessi
- OPS03-BP02 Potere di intervento dei membri del team quando i risultati sono a rischio
- OPS03-BP03 Incoraggiamento all'escalation
- OPS03-BP07 Fornitura di risorse appropriate ai team
- OPS09-BP01 Misura gli obiettivi operativi e i KPI con le metriche
- OPS09-BP03 Revisione delle metriche operative e assegnazione delle priorità per favorire il miglioramento
- OPS11-BP01 Definizione di un processo per il miglioramento continuo

Documenti correlati:

- Whitepaper AWS - Introduzione a DevOps in AWS
- Whitepaper AWS - Cloud AWS Adoption Framework: Operations Perspective
- Eccellenza operativa del Framework AWS Well-Architected: topologie del modello operativo a livello di carico di lavoro
- AWS Prescriptive Guidance - Building your Cloud Operating Model
- AWS Prescriptive Guidance - Create a RACI or RASCI matrix for a cloud operating model
- Blog sulle operazioni e le migrazioni Cloud AWS - Delivering Business Value with Cloud Platform Teams

- Blog sulle operazioni e le migrazioni Cloud AWS - Why a Cloud Operating Model?
- Blog AWS DevOps - How organizations are modernizing for cloud operations

Video correlati:

- AWS Summit Online - Cloud Operating Models for Accelerated Transformation
- AWS re:Invent 2023 - Future-proofing cloud security: A new operating model

OPS02-BP05 Definizione di meccanismi per richiedere aggiunte, modifiche ed eccezioni

È possibile effettuare richieste ai titolari di processi, procedure e risorse. Tra le richieste figurano aggiunte, modifiche ed eccezioni. Tali richieste passano attraverso un processo di gestione delle modifiche Prendi decisioni informate per approvare le richieste quando vengono ritenute fattibili e appropriate dopo una valutazione dei vantaggi e dei rischi.

Risultato desiderato:

- Puoi effettuare richieste per modificare processi, procedure e risorse sulla base della titolarità assegnata.
- Le modifiche vengono eseguite in modo deliberato, valutando benefici e rischi.

Anti-pattern comuni:

- Devi aggiornare il modo di implementare la tua applicazione, ma non esiste un metodo per richiedere una modifica al processo di implementazione al team operativo.
- Il piano di disaster recovery deve essere aggiornato, ma non è stato identificato il proprietario a cui richiedere le modifiche.

Vantaggi dell'adozione di questa best practice:

- Processi, procedure e risorse possono evolvere mentre cambiano i requisiti.
- I titolati possono prendere decisioni mirate su quando effettuare le modifiche.
- Le modifiche vengono eseguite deliberatamente.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Per implementare questa best practice devi essere in grado di richiedere modifiche a processi, procedure e risorse. Il processo di gestione delle modifiche può essere semplice. Documenta il processo di gestione delle modifiche.

Esempio del cliente

AnyCompany Retail usa una matrice di assegnazione delle responsabilità (RACI) per identificare il proprietario delle modifiche per processi, procedure e risorse. L'azienda dispone di un processo documentato di gestione delle modifiche, semplice e facile da seguire. Tramite il processo e la matrice RACI, tutti possono inviare richieste di modifiche.

Passaggi dell'implementazione

1. Identifica i processi, le procedure e le risorse per il tuo carico di lavoro e i proprietari di ciascun elemento. Documentali nel tuo sistema di gestione delle conoscenze.
   a. In caso di mancata implementazione, inizia da OPS02-BP01 Le risorse hanno identificato i proprietari, OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure o OPS02-BP03 Assegnazione di proprietari identificati alle operazioni che siano responsabili delle relative prestazioni.
2. Collabora con le parti interessate all'interno della tua azienda per sviluppare un processo di gestione delle modifiche. Il processo deve includere aggiunte, modifiche ed eccezioni per risorse, processi e procedure.
   a. Puoi utilizzare AWS Systems Manager Change Manager come piattaforma di gestione delle modifiche per le risorse del carico di lavoro.
3. Documenta il processo di gestione delle modifiche nel tuo sistema di gestione delle conoscenze.

Livello di impegno per il piano di implementazione: medio Sviluppare un processo di gestione delle modifiche significa garantire un allineamento con più parti interessate all'interno dell'organizzazione.

Risorse

Best practice correlate:

- OPS02-BP01 Le risorse hanno identificato i proprietari: le risorse richiedono proprietari identificati prima di creare un processo di gestione delle modifiche.
- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure: i processi richiedono proprietari identificati prima di creare un processo di gestione delle modifiche.

- OPS02-BP03 Assegnazione di proprietari identificati alle operazioni che siano responsabili delle relative prestazioni: le attività di operazioni richiedono proprietari identificati prima di creare un processo di gestione delle modifiche.

Documenti correlati:

- AWS Prescriptive Guidance, playbook di base per migrazioni di AWS grandi dimensioni: creazione di matrici RACI
- Whitepaper sulla gestione delle modifiche nel cloud

Servizi correlati:

- AWS Systems Manager Change Manager

OPS02-BP06 Predefinizione o negoziazione delle responsabilità tra i team

Predisponi accordi definiti o concordati tra i team che descrivono come funzionano e si supportano reciprocamente (ad esempio, tempi di risposta, obiettivi o contratti relativi al livello di servizio). I canali di comunicazione tra team sono documentati. Comprendere l'impatto del lavoro dei team sui risultati aziendali e sui risultati di altri team e organizzazioni fornisce indicazioni in merito alla priorità dei loro compiti e consente loro di rispondere in modo appropriato.

Quando la responsabilità e la proprietà sono indefinite o sconosciute, rischi di non affrontare le attività necessarie in modo tempestivo e di impiegare sforzi ridondanti e potenzialmente conflittuali per rispondere a tali esigenze.

Risultato desiderato:

- Il lavoro tra team o gli accordi di assistenza vengono concordati e documentati.
- I team che supportano o lavorano con altri hanno definito i canali di comunicazione e le aspettative in termini di risposte.

Anti-pattern comuni:

- In produzione si verifica un problema e due team separati iniziano a cercare la soluzione senza confrontarsi. Il loro impegno separato prolunga l'interruzione.

- Il team operativo ha bisogno di assistenza dal team di sviluppo, ma non c'è un accordo sui tempi di risposta. La richiesta si blocca nel backlog.

Vantaggi dell'adozione di questa best practice:

- I team sanno come interagire e supportarsi a vicenda.
- Le aspettative relative ai tempi di risposta sono note.
- I canali di comunicazione sono definiti in modo chiaro.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all’implementazione

Se si implementa questa best practice non ci saranno dubbi sulla collaborazione tra team. Gli accordi formali codificano il modo di collaborare o di assistersi a vicenda dei team. I canali di comunicazione tra team sono documentati.

Esempio del cliente

Il team SRE di AnyCompany Retail ha un contratto sul livello di servizio (SLA) con il team di sviluppo. Ogni volta che il team di sviluppo effettua una richiesta nel sistema di ticketing, riceve una risposta entro 15 minuti. Se si verifica un malfunzionamento presso la sede, il team SRE assume il comando delle indagini con il supporto del team di sviluppo.

Passaggi dell'implementazione

1. Collaborando con le parti interessate all'interno dell'organizzazione, sviluppa accordi tra team basati su processi e procedure.
   a. Se i due team condividono un processo o una procedura, crea un runbook sulle modalità di collaborazione dei team.
   b. Se esistono dipendenze tra i team, concorda uno SLA per le risposte alle richieste.
2. Inserisci le responsabilità nel tuo sistema di gestione delle conoscenze.

Livello di impegno per il piano di implementazione: medio Se non esistono accordi tra i team, può essere impegnativo raggiungere un accordo con le parti interessate all'interno dell'organizzazione.

Risorse

Best practice correlate:

- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure: la proprietà del processo deve essere identificata prima di stabilire accordi tra i team.
- OPS02-BP03 Assegnazione di proprietari identificati alle operazioni che siano responsabili delle relative prestazioni: la proprietà delle operazioni deve essere identificata prima di stabilire accordi tra i team.

Documenti correlati:

- AWS Executive Insights - Empowering Innovation with the Two-Pizza Team
- Introduction to DevOps on AWS - Two-Pizza Teams

OPS 3. In che modo la cultura aziendale supporta i risultati aziendali?

Fornisci supporto ai membri del team in modo che possano essere più efficaci nell'azione e nel supporto dei risultati aziendali.

Best practice

- OPS03-BP01 Definizione della sponsorizzazione esecutiva
- OPS03-BP02 Potere di intervento dei membri del team quando i risultati sono a rischio
- OPS03-BP03 L'escalation è incoraggiata
- OPS03-BP04 Comunicazioni tempestive, chiare e fruibili
- OPS03-BP05 Incoraggiamento alla sperimentazione
- OPS03-BP06 Incoraggiamento ai membri del team a mantenere e ampliare le proprie competenze
- OPS03-BP07 Team di risorse appropriati

OPS03-BP01 Definizione della sponsorizzazione esecutiva

Ai massimi livelli, gli alti dirigenti fungono da sponsor esecutivo per definire chiaramente le aspettative e la direzione dei risultati dell'organizzazione, compresa la valutazione del successo. Lo sponsor sostiene e promuove l'adozione delle best practice e l'evoluzione dell'organizzazione.

Risultato desiderato: definizione di linee chiare in termini di leadership e responsabilità per i risultati desiderati da parte delle organizzazioni impegnate nell'adottare, trasformare e ottimizzare le proprie operazioni cloud. L'organizzazione comprende ogni capacità richiesta per raggiungere un nuovo risultato e assegna la proprietà ai team funzionali per lo sviluppo. La leadership implementa attivamente questa direzione, assegna la proprietà, si assume la responsabilità e definisce il lavoro. Di conseguenza, le persone in tutta l'organizzazione possono mobilitarsi, sentirsi ispirate e lavorare attivamente per raggiungere gli obiettivi desiderati.

Anti-pattern comuni:

- I proprietari dei carichi di lavoro sono tenuti a migrare i carichi di lavoro su AWS senza uno sponsor e un piano chiari per le operazioni cloud. I team pertanto non collaborano in modo consapevole per migliorare e consolidare le proprie capacità operative. La mancanza di standard operativi sulle best practice mette in difficoltà i team, ad esempio il lavoro degli operatori, le chiamate e il debito tecnico, limitando l'innovazione.
- È stato fissato un nuovo obiettivo a livello di organizzazione per adottare una tecnologia emergente senza fornire sponsor e strategia di leadership. I team interpretano gli obiettivi in modo diverso, il che crea confusione su dove concentrare gli impegni, sul perché sono importanti e su come misurare l'impatto. Di conseguenza, l'organizzazione perde slancio nell'adozione della tecnologia.

Vantaggi dell'adozione di questa best practice: se lo sponsor esecutivo comunica e condivide in modo chiaro visione, direzione e obiettivi, i membri del team conoscono le aspettative riposte su di loro. Quando i leader sono coinvolti attivamente, le persone e i team iniziano a concentrare attivamente gli impegni nella stessa direzione per raggiungere gli obiettivi definiti. L'organizzazione di conseguenza massimizza la capacità di successo. Quando si valuta il successo, è possibile identificare meglio gli ostacoli al suo conseguimento in modo da affrontarli attraverso l'intervento dello sponsor esecutivo.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

- In ogni fase del percorso verso il cloud (migrazione, adozione oppure ottimizzazione), il successo richiede un coinvolgimento attivo ai massimi livelli della leadership con uno sponsor esecutivo designato. Lo sponsor esecutivo allinea la mentalità, le competenze e le modalità di lavoro del team alla strategia definita.
  - Spiega il perché: chiarisci e illustra il ragionamento alla base di visione e strategia.

- Definisci le aspettative: definisci e pubblica gli obiettivi per le tue organizzazioni, incluso il modo in cui verranno misurati.
- Tieni traccia del conseguimento degli obiettivi: misura con regolarità il conseguimento incrementale degli obiettivi (non solo il completamento delle attività). Condividi i risultati in modo da poter intraprendere le azioni appropriate se si evidenziano dei rischi.
- Fornisci le risorse necessarie per raggiungere gli obiettivi: favorisci la collaborazione tra persone e team al fine di sviluppare le soluzioni giuste che garantiscano i risultati definiti. Ciò riduce o elimina gli attriti organizzativi.
- Sostieni i team: mantieni un coinvolgimento attivo con i tuoi team in modo da comprenderne le prestazioni e l'eventuale presenza di fattori di influenza esterni. Individua gli ostacoli che impediscono i progressi dei team. Agisci per conto dei tuoi team per superare gli ostacoli e rimuovere gli oneri superflui. Quando i team sono influenzati da fattori esterni, rivaluta gli obiettivi e modifica i target in base alle esigenze.
- Promuovi l'adozione delle best practice: riconosci le best practice che offrono vantaggi quantificabili e identifica creatori e destinatari. Incoraggia ulteriormente l'adozione per amplificare i vantaggi ottenuti.
- Incoraggia l'evoluzione dei team: crea una cultura di miglioramento continuo e impara in modo proattivo da progressi e insuccessi. Incoraggia la crescita e lo sviluppo sia personale sia organizzativo. Usa dati e aneddoti per migliorare la visione e la strategia.

Esempio del cliente

AnyCompany Retail è in fase di trasformazione aziendale attraverso il rapido rinnovamento delle esperienze dei clienti, il miglioramento della produttività e l'accelerazione della crescita con l'IA generativa.

Passaggi dell'implementazione

1. Stabilisci una leadership a thread singolo e assegna uno sponsor esecutivo principale per guidare e gestire la trasformazione.
2. Definisci chiaramente i risultati aziendali della trasformazione e assegna proprietà e responsabilità. Fornisci allo sponsor esecutivo principale l'autorità di guidare e prendere decisioni critiche.
3. Verifica che la strategia di trasformazione sia stata definita molto chiaramente e ampiamente comunicata dallo sponsor esecutivo a tutti i livelli dell'organizzazione.
   a. Definisci chiaramente gli obiettivi aziendali per le iniziative IT e cloud.

   b. Documenta le principali metriche aziendali per promuovere la trasformazione dell'IT e del cloud.
   c. Comunica la visione in modo coerente a tutti i team e alle persone responsabili di parti della strategia.
4. Sviluppa matrici di pianificazione della comunicazione che specifichino quale messaggio deve essere recapitato a leader, manager e singoli collaboratori specifici. Specifica la persona o il team che deve recapitare questo messaggio.
   a. Rispetta i piani di comunicazione in modo coerente e affidabile.
   b. Stabilisci e gestisci le aspettative attraverso eventi di persona su base regolare.
   c. Accetta il feedback sull'efficacia delle comunicazioni, quindi modifica le comunicazioni e pianifica di conseguenza.
   d. Pianifica gli eventi di comunicazione per comprendere in modo proattivo le sfide dei team e stabilire un ciclo di feedback coerente che consenta di correggere la direzione laddove necessario.
5. Coinvolgi in modo attivo ogni iniziativa dal punto di vista della leadership per verificare che tutti i team interessati comprendano i risultati di cui sono responsabili.
6. In ogni riunione sullo stato, gli sponsor esecutivi devono individuare gli ostacoli, esaminare metriche, aneddoti o feedback dei team nonché misurare i progressi verso il raggiungimento degli obiettivi.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS03-BP04 Comunicazioni tempestive, chiare e fruibili
- OPS11-BP01 Definizione di un processo per il miglioramento continuo
- OPS11-BP07 Revisione dei parametri delle operazioni

Documenti correlati:

- Untangling Your Organisational Hairball: Highly Aligned
- The Living Transformation: Pragmatically approaching changes
- Becoming a Future-Ready Enterprise
- 7 Pitfalls to Avoid When Building a CCOE

- Navigating the Cloud: Key Performance Indicators for Success

Video correlati:

- AWS re:Invent 2023: A leader's guide to generative AI: Using history to shape the future (SEG204)

Esempi correlati:

- Prosci: Primary Sponsor's Role & Importance

OPS03-BP02 Potere di intervento dei membri del team quando i risultati sono a rischio

Il comportamento culturale della responsabilità instillato dalla leadership fa sì che ogni dipendente si senta autorizzato ad agire per conto dell'intera azienda, al di là del proprio ambito definito da ruolo e responsabilità. I dipendenti possono intervenire per identificare in modo proattivo i rischi man mano che emergono e intraprendere le azioni appropriate. Tale cultura consente ai dipendenti di prendere decisioni di alto valore in quanto consapevoli della situazione.

Ad esempio, Amazon utilizza i principi di leadership come linee guida per agevolare comportamenti migliori dei dipendenti nelle situazioni, la risoluzione dei problemi, l'affrontare i conflitti e l'agire.

Risultato desiderato: una nuova cultura stabilita dalla leadership che consente a persone e di prendere decisioni critiche, anche ai livelli inferiori dell'organizzazione (a condizione che le decisioni a lungo termine siano definite con autorizzazioni e meccanismi di sicurezza sottoponibili ad audit). L'errore non è una mancanza, i team imparano in modo iterativo a migliorare il processo decisionale e le risposte per affrontare situazioni simili in futuro. Se le azioni già intraprese portano a un miglioramento che può avvantaggiare altri team, occorre condividere in modo proattivo le conoscenze derivanti da tali azioni. La leadership misura i miglioramenti operativi e incentiva le persone e l'organizzazione all'adozione di tali modelli.

Anti-pattern comuni:

- Nell'organizzazione non esistono linee guida o meccanismi chiari su cosa fare quando viene identificato un rischio. Ad esempio, quando un dipendente nota un attacco di phishing, non lo segnala al team di sicurezza, con il risultato che gran parte dell'organizzazione è vittima dell'attacco, causando una violazione dei dati.
- I clienti si lamentano dell'indisponibilità del servizio, che deriva principalmente da implementazioni non riuscite. Il team SRE è responsabile dello strumento di implementazione e il rollback

automatico per le implementazioni è nella roadmap a lungo termine. In un recente rollout dell'applicazione, uno degli ingegneri ha fornito una soluzione per automatizzare il ripristino dell'applicazione a una versione precedente. Sebbene la soluzione possa diventare il modello per i team SRE, altri team non la adottano poiché non esiste un processo per monitorare i miglioramenti. L'organizzazione continua a essere afflitta da implementazioni non corrette che hanno un impatto sui clienti e provocano ulteriore insoddisfazione.

- Per rispettare la conformità, il team di infosec supervisiona un processo consolidato per ruotare regolarmente le chiavi SSH condivise per conto degli operatori che si connettono alle loro istanze Amazon EC2 Linux. I team di infosec impiegano diversi giorni per completare la rotazione delle chiavi e la connessione alle istanze viene bloccata. Nessuno all'interno o all'esterno di infosec suggerisce di utilizzare altre opzioni su AWS per ottenere lo stesso risultato.

Vantaggi dell'adozione di questa best practice: la decentralizzazione dell'autorità per l'adozione delle decisioni e la concessione ai team della possibilità di adottare decisioni chiave consente di affrontare i problemi più rapidamente, con percentuali di successo crescenti. Inoltre, i team iniziano a percepire un senso di appartenenza e gli errori sono accettabili. La sperimentazione diventa un pilastro culturale. Manager e direttori non si sentono controllati in ogni aspetto del loro lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

1. Sviluppa una cultura che preveda il verificarsi errori.
2. Definisci chiaramente proprietà e responsabilità per le varie aree funzionali all'interno dell'organizzazione.
3. Comunica la proprietà e la responsabilità a tutti in modo che le persone sappiano chi può facilitare le decisioni decentralizzate.
4. Stabilisci le decisioni definitive e reversibili per permettere alle persone di sapere quando è necessario eseguire l'escalation a livelli più alti di leadership.
5. Crea la consapevolezza organizzativa secondo cui tutti i dipendenti hanno la capacità di agire a vari livelli quando i risultati sono a rischio. Fornisci ai membri del team la documentazione sulla governance, i livelli di autorizzazione, gli strumenti e le opportunità per mettere in pratica le competenze necessarie e intervenire in modo efficace.
6. Offri ai membri del team l'opportunità di mettere in pratica le competenze necessarie per rispondere a varie decisioni. Una volta definiti i livelli decisionali, organizza delle giornate di gioco per verificare che tutti i singoli collaboratori comprendano e possano usare il processo.

   a. Fornisci ambienti sicuri alternativi in cui testare i processi e sottoporre i membri del team alla dovuta formazione.
   b. Riconosci e crea la consapevolezza secondo cui i membri del team hanno l'autorità di agire quando il risultato ha un livello di rischio prestabilito.
   c. Definisci l'autorità dei membri del team per intervenire assegnando le autorizzazioni e l'accesso ai carichi di lavoro e ai componenti supportati.
7. Offri ai team la possibilità di condividere le proprie conoscenze (successi e fallimenti operativi).
8. Consenti ai team di sfidare lo status quo e fornisci i meccanismi per monitorare e misurare i miglioramenti, nonché il loro impatto sull'organizzazione.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS01-BP06 Valutazione dei compromessi gestendo vantaggi e rischi
- OPS02-BP05 Definizione di meccanismi per identificare responsabilità e proprietà

Documenti correlati:

- Post sul blog AWS | The agile enterprise
- Post sul blog AWS | Measuring success: A paradox and a plan
- Post sul blog AWS | Letting go: Enabling autonomy in teams
- Centralize or Decentralize?

Video correlati:

- re:Invent 2023 | How to not sabotage your transformation (SEG201)
- re:Invent 2021 | Amazon Builders' Library: Operational Excellence at Amazon
- Centralization vs. Decentralization

Esempi correlati:

- Using architectural decision records to streamline technical decision-making for a software development project

OPS03-BP03 L'escalation è incoraggiata

I membri del team sono incoraggiati dalla leadership a segnalare problemi e preoccupazioni ai responsabili delle decisioni e alle parti interessate di alto livello se ritengono che i risultati desiderati siano a rischio e gli standard previsti non siano rispettati. Questa è una funzionalità della cultura dell'organizzazione ed è implementata a tutti i livelli. L'escalation deve essere eseguita in anticipo e di frequente, in modo da identificare i rischi e limitarli prima che provochino incidenti. La leadership non rimprovera le persone per aver effettuato l'escalation di un problema.

Risultato desiderato: possibilità per le persone in tutta di eseguire l'escalation dei problemi ai loro livelli di leadership immediati e superiori. La leadership ha stabilito deliberatamente e consapevolmente l'aspettativa che i propri team si sentano tranquilli nell'eseguire l'escalation di qualsiasi problema. Esiste un meccanismo per eseguire l'escalation dei problemi a ogni livello dell'organizzazione. Quando un dipendente esegue l'escalation al proprio manager, insieme decidono il livello di impatto e se il problema debba essere ulteriormente scalato. Per iniziare l'escalation, i dipendenti sono tenuti a includere un piano di lavoro consigliato per risolvere il problema. Se la direzione non interviene tempestivamente, i dipendenti sono incoraggiati a inoltrare i problemi al massimo livello di leadership se ritengono fermamente che i rischi per l'organizzazione giustifichino l'escalation.

Anti-pattern comuni:

- I dirigenti non pongono domande approfondite durante la riunione sullo stato del programma di trasformazione del cloud per scoprire dove si verificano problemi e ostacoli. Solo le buone notizie vengono presentate nello stato. The CIO ha chiarito che le piace solo sentire buone notizie, poiché qualsiasi sfida sollevata fa CEO pensare che il programma stia fallendo.
- Sei un ingegnere delle operazioni cloud e noti che il nuovo sistema di gestione delle conoscenze non è ampiamente adottato dai team applicativi. L'azienda ha investito un anno di tempo e diversi milioni di dollari per implementare questo nuovo sistema di gestione delle conoscenze, ma le persone continuano a creare i propri runbook localmente e a condividerli su una condivisione cloud aziendale, rendendo difficile l'individuazione delle conoscenze pertinenti ai carichi di lavoro supportati. Cerchi di portare questo aspetto all'attenzione della dirigenza perché l'uso coerente del sistema può migliorare l'efficienza operativa. Quando lo comunichi alla direttrice a capo dell'implementazione del sistema di gestione delle conoscenze, ti rimprovera perché tale aspetto mette in discussione l'investimento.

- Il team di infosec responsabile del rafforzamento delle risorse di elaborazione ha deciso di mettere in atto un processo che richiede l'esecuzione delle scansioni necessarie per garantire che le EC2 istanze siano completamente protette prima che il team di elaborazione rilasci la risorsa per l'uso. Ciò ha comportato un ritardo di un'ulteriore settimana per l'implementazione delle risorse, il che interrompe il loro periodo di tempo. SLA Il team di calcolo non desidera inoltrare la questione al vicepresidente tramite cloud perché ciò mette in cattiva luce il vicepresidente della sicurezza delle informazioni.

Vantaggi dell'adozione di questa best practice:

I problemi complessi o critici vengono risolti prima che abbiano impatto sull'azienda. Si perde meno tempo. I rischi sono ridotti al minimo. I team diventano più proattivi e concentrati sui risultati della risoluzione dei problemi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

La volontà e la capacità di crescere liberamente a tutti i livelli dell'organizzazione sono la base organizzativa e culturale da sviluppare consapevolmente attraverso una formazione appropriata, le comunicazioni della leadership, la definizione delle aspettative e l'implementazione di meccanismi a tutti i livelli dell'organizzazione.

Passaggi dell'implementazione

1. Definisci policy, standard e aspettative per l'organizzazione.
   a. Garantisci un'ampia adozione e comprensione delle policy, delle aspettative e degli standard.
2. Incoraggia, forma e responsabilizza i lavoratori a eseguire un'escalation anticipata e frequente quando gli standard non vengono rispettati.
3. Riconosci a livello organizzativo che l'escalation anticipata e frequente è la best practice. Accetti che le escalation possono rivelarsi infondate e che è meglio avere l'opportunità di prevenire un incidente piuttosto che privarsi di quell'opportunità senza escalation.
   a. Predisponi un meccanismo di escalation (come un sistema Andon cord).
   b. È opportuno disporre di procedure documentate che definiscano quando e come deve verificarsi l'escalation.
   c. Definisci la serie di persone in ordine di autorità cui è consentito intraprendere o approvare azioni, nonché le informazioni di contatto di ciascuna parte interessata.

4. Un'escalation deve continuare fino a quando il membro del team non è convinto che il rischio sia stato mitigato attraverso le azioni guidate dalla leadership.
   a. Le escalation devono includere:
      i. la descrizione della situazione e la natura del rischio;
      ii. le criticità della situazione;
      iii. chi o cosa è interessato;
      iv. il livello dell'impatto;
      v. l'urgenza in caso di impatto;
      vi. i rimedi suggeriti e i piani di mitigazione.
   b. Proteggi i dipendenti coinvolti nell'escalation. È necessario predisporre una policy che protegga i membri del team da eventuali ritorsioni se si trovano a dover scavalcare una parte interessata o un responsabile delle decisioni non reattivo. Metti in atto dei meccanismi per identificare se ciò si verifica e rispondere in modo appropriato.
5. Incoraggia la cultura del miglioramento continuo e dei cicli di feedback in tutto ciò che l'organizzazione produce. I cicli di feedback fungono da piccole escalation per le persone responsabili e identificano le opportunità di miglioramento, anche quando l'escalation non è necessaria. La cultura del miglioramento continuo obbliga tutti a essere più proattivi.
6. La leadership deve periodicamente ribadire le policy, gli standard, i meccanismi e il desiderio di un'escalation aperta e di cicli di feedback continui senza penalità.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS02-BP05 Definizione di meccanismi per richiedere aggiunte, modifiche ed eccezioni

Documenti correlati:

- How do you foster a culture of continuous improvement and learning from Andon and escalation systems?
- The Andon Cord (IT Revolution)
- AWS DevOps Linee guida | Stabilisci percorsi di escalation chiari e incoraggia un disaccordo costruttivo

Video correlati:

- Jeff Bezos on how to make decisions (& increase velocity)
- Toyota Product System: Stopping Production, a Button, and an Andon Electric Board
- Andon Cord nel settore della produzione LEAN

Esempi correlati:

- Working with escalation plans in Incident Manager

OPS03-BP04 Comunicazioni tempestive, chiare e fruibili

La leadership è responsabile della creazione di comunicazioni forti ed efficaci, soprattutto quando l'organizzazione adotta nuove strategie, tecnologie o modalità di lavoro. I leader devono stabilire le aspettative affinché tutto il personale lavori per raggiungere gli obiettivi aziendali. Elabora meccanismi di comunicazione che creino e mantengano la consapevolezza tra i team responsabili della gestione dei piani finanziati e sponsorizzati dalla leadership. Utilizza la diversità interorganizzativa e ascolta con attenzione i vari punti di vista. Usa questa prospettiva per incrementare l'innovazione, mettere in discussione le tue ipotesi e ridurre il rischio di bias confermativi. Favorisci l'inclusione, la diversità e l'accessibilità all'interno dei team per ottenere prospettive vantaggiose.

Risultato desiderato: elaborazione di strategie di comunicazione da parte della tua organizzazione per gestire l'impatto del cambiamento sull'organizzazione. I team sono informati e motivati a continuare a lavorare insieme anziché l'uno contro l'altro. Le persone comprendono quanto sia importante il proprio ruolo per raggiungere gli obiettivi stabiliti. L'e-mail è solo un meccanismo passivo per le comunicazioni e viene utilizzato di conseguenza. La direzione trascorre tempo con i singoli collaboratori per aiutarli a comprendere le proprie responsabilità, le attività da completare e in che modo il loro lavoro contribuisce alla missione generale. Quando necessario, i leader coinvolgono direttamente le persone in un ambiente più piccolo per trasmettere il messaggio e verificare che venga recepito in modo efficace. Come risultato di buone strategie di comunicazione, l'organizzazione si comporta in misura pari o superiore alle aspettative della leadership. La leadership incoraggia e desidera esaminare opinioni diverse all'interno dell'organizzazione e tra i team.

Anti-pattern comuni:

- L'organizzazione ha un piano quinquennale per migrare tutti i carichi di lavoro su AWS. Il business case per il cloud include la modernizzazione del 25% di tutti i carichi di lavoro per utilizzare la tecnologia serverless. Il CIO comunica questa strategia ai collaboratori diretti e si aspetta che

ogni leader trasmetta questa presentazione a manager, direttori e singoli collaboratori senza comunicazioni di persona. Il CIO fa un passo indietro e si aspetta che l'organizzazione esegua la nuova strategia.

- La leadership non fornisce né utilizza un meccanismo di feedback e aumenta il divario nelle aspettative, causando lo stallo dei progetti.
- Ti viene chiesto di apportare una modifica ai gruppi di sicurezza, ma non ricevi i dettagli sulle stesse, sull'impatto della modifica su tutti i carichi di lavoro e sulla data della modifica. Il manager inoltra un'e-mail del vicepresidente di infosec e aggiunge il messaggio "Fai in modo che accada".
- Sono state apportate modifiche alla strategia di migrazione che riducono la percentuale di modernizzazione pianificata dal 25% al 10%. La riduzione ha effetti a valle sull'organizzazione delle operazioni. Questo cambiamento strategico non è stato comunicato e quindi non è disponibile la capacità qualificata sufficiente per supportare un numero maggiore di carichi di lavoro in lift and shift in AWS.

Vantaggi dell'adozione di questa best practice:

- L'organizzazione è ben informata sulle strategie nuove o modificate e agisce di conseguenza con una forte motivazione alla collaborazione per raggiungere gli obiettivi e le metriche generali stabiliti dalla leadership.
- Esistono meccanismi utilizzati per fornire tempestivamente notifiche ai membri del team in merito a rischi noti ed eventi pianificati.
- Le nuove modalità di lavoro, compresi i cambiamenti relativi a personale o organizzazione, processi o tecnologia, insieme alle competenze richieste, vengono adottate in modo più efficace dall'organizzazione che quindi realizza i vantaggi aziendali più rapidamente.
- I membri del team hanno il contesto necessario per ricevere le comunicazioni e possono essere più efficaci nel loro lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Per implementare questa best practice, devi collaborare con le parti interessate presenti nell'organizzazione per concordare gli standard di comunicazione. Comunica tali standard alla tua organizzazione. Per qualsiasi transizione IT significativa, il team di pianificazione definito può gestire con maggiore successo l'impatto del cambiamento sulle persone rispetto a un'organizzazione che ignora questa procedura. La gestione del cambiamento può essere impegnativa per le organizzazioni

poiché richiede un forte consenso sulla nuova strategia di tutti i singoli collaboratori. In assenza di un team di pianificazione della transizione, la leadership ha il 100% della responsabilità di condurre comunicazioni efficaci. Quando si crea un team di pianificazione della transizione, comunica ai membri del team di collaborare con tutta la leadership organizzativa per definire e gestire comunicazioni efficaci a tutti i livelli.

Esempio del cliente

AnyCompany Retail si è registrata al supporto Enterprise AWS e dipende da altri fornitori di terze parti per le operazioni cloud. L'azienda utilizza chat e chatop come principale mezzo di comunicazione per le attività operative. Allarmi e altre informazioni caratterizzano canali specifici. Quando qualcuno deve intervenire, il risultato desiderato viene definito in modo chiaro e, in molti casi, la persona riceve un runbook o un playbook da usare. Viene utilizzato un calendario delle modifiche per pianificare i cambiamenti più importanti ai sistemi di produzione.

Passaggi dell'implementazione

1. Crea un team principale all'interno dell'organizzazione che abbia la responsabilità di elaborare e avviare i piani di comunicazione dei cambiamenti che avvengono a più livelli all'interno dell'organizzazione.
2. Istituisci la proprietà a thread singolo per la supervisione. Offri ai singoli team la capacità di innovare in modo indipendente e bilanciare l'uso di meccanismi coerenti, consentendo così il giusto livello di ispezione e visione della direzione.
3. Collabora con le parti interessate di tutta l'organizzazione per concordare standard, procedure e piani di comunicazione.
4. Verifica che il team di comunicazione principale collabori con la leadership dell'organizzazione e del programma per creare messaggi per il personale appropriato per conto dei leader.
5. Sviluppa meccanismi di comunicazione strategici per gestire il cambiamento attraverso annunci, calendari condivisi, riunioni plenarie e metodi di persona o individuali, in modo che i membri del team abbiano le giuste aspettative sulle azioni da intraprendere.
6. Quando possibile, comunica contesto, dettagli e tempo necessari per determinare se è richiesta un'azione. Quando è necessaria un'azione, indica l'azione richiesta e il suo impatto.
7. Implementa strumenti che agevolino le comunicazioni tattiche, come chat interna, e-mail e gestione delle conoscenze.
8. Implementa meccanismi per misurare e verificare che tutte le comunicazioni portino ai risultati desiderati.

9. Stabilisci un ciclo di feedback che misuri l'efficacia delle comunicazioni, specialmente quando sono correlate alla resistenza ai cambiamenti nell'organizzazione.
10. Per tutti gli Account AWS, stabilisci contatti alternativi per fatturazione, sicurezza e operazioni. Idealmente, ogni contatto deve essere una distribuzione di e-mail anziché una comunicazione individuale specifica.
11. Stabilisci un piano di comunicazione di escalation e annullamento dell'escalation per interagire con i team interni ed esterni, compreso il supporto AWS e altri fornitori di terze parti.
12. Avvia ed esegui le strategie di comunicazione in modo coerente per tutta la durata di ciascun programma di trasformazione.
13. Assegna le priorità alle azioni ripetibili, ove possibile, per automatizzarle in sicurezza su larga scala.
14. Quando le comunicazioni sono richieste in scenari con azioni automatizzate, lo scopo della comunicazione deve essere informare i team, per il controllo o una parte del processo di gestione delle modifiche.
15. Analizza le comunicazioni provenienti dai sistemi di avviso per individuare i falsi positivi o gli avvisi creati costantemente. Rimuovi o modifica questi avvisi in modo che vengano inviati quando è richiesto l'intervento umano. Se viene attivato un avviso, fornisci un runbook o un playbook.
    a. Puoi affidarti ai documenti di AWS Systems Manager per creare playbook e runbook per gli avvisi.
16. Sono stati attivati meccanismi per fornire tempestivamente notifiche in merito ai rischi o agli eventi pianificati in modo chiaro e fruibile al fine di consentire risposte appropriate. Usa elenchi di indirizzi e-mail o canali di chat per inviare le notifiche di preavviso rispetto agli eventi pianificati.
    a. Puoi usare AWS Chatbot per inviare avvisi e rispondere agli eventi all'interno della piattaforma di messaggistica della tua organizzazione.
17. Fornisci una fonte di informazioni accessibile dove è possibile individuare gli eventi pianificati. Fornisci le notifiche degli eventi pianificati dallo stesso sistema.
    a. AWS Systems Manager Change Calendar consente di creare finestre di modifica in cui queste possono verificarsi. In questo modo i membri del team ricevono un preavviso su quando poter effettuare la modifica in modo sicuro.
18. Monitora le notifiche di vulnerabilità e le informazioni sulle patch per capire le vulnerabilità in circolazione e i rischi potenziali associati ai componenti del tuo carico di lavoro. Invia notifiche ai membri del team in modo che possano intervenire.
    a. Puoi iscriverti ai bollettini sulla sicurezza AWS per ricevere notifiche relative a vulnerabilità su AWS.

19. Cerca opinioni e prospettive diverse: incoraggia la condivisione dei contributi da parte di tutti. Offri opportunità di comunicazione ai gruppi sottorappresentati. Distribuisci a rotazione i ruoli e le responsabilità nelle riunioni.
    a. Amplia ruoli e responsabilità: offri ai membri del team l'opportunità di assumere ruoli che altrimenti potrebbero altrimenti non ricoprire mai. Ciò consentirà loro di acquisire esperienza e nuove prospettive grazie anche alle interazioni con i nuovi membri del team, con i quali potrebbero non interagire altrimenti. Un mutuo scambio di esperienze e punti di vista vantaggioso per tutti. Con l'aumento delle prospettive, identifica le opportunità aziendali emergenti o le nuove opportunità di miglioramento. Fai in modo che i membri di un team svolgano a turno attività comuni eseguite normalmente da altri affinché comprendano richieste e impatto delle loro prestazioni.
    b. Garantisci un ambiente sicuro e ospitale: adotta policy e controlli che consentano di proteggere la sicurezza fisica e mentale dei membri del team all'interno dell'organizzazione. I membri del team devono poter interagire senza alcun timore. Quando i membri del team si sentono al sicuro e ben accolti, è più probabile che siano coinvolti e produttivi. Più è diversificata la tua organizzazione, migliore sarà la comprensione nei confronti delle persone supportate, compresi i clienti. Quando i membri del team si sentono a loro agio, sono liberi di parlare e sono sicuri di essere ascoltati, con maggiori probabilità condivideranno approfondimenti preziosi (ad esempio, opportunità di marketing, esigenze di accessibilità, segmenti di mercato non serviti, rischi non riconosciuti nel tuo ambiente).
    c. Consenti la totale partecipazione dei membri del team: fornisci le risorse necessarie ai dipendenti affinché partecipino appieno a tutte le attività correlate al lavoro. I membri del team che affrontano sfide quotidiane hanno sviluppato competenze per superarle. Queste competenze esclusive possono offrire vantaggi significativi alla tua organizzazione. Grazie al supporto di strutture adeguate, i membri del team possono apportare contributi vantaggiosi.

Risorse

Best practice correlate:

- OPS03-BP01 Definizione della sponsorizzazione esecutiva
- OPS07-BP03 Utilizzo di runbook per eseguire le procedure
- OPS07-BP04 Utilizzo dei playbook per analizzare i problemi

Documenti correlati:

- Post sul blog AWS | Accountability and empowerment are key to high-performing agile organizations
- AWS Executive Insights | Learn to scale innovation, not complexity | Single-threaded Leaders
- AWS Security Bulletins
- Open CVE
- Supporto App in Slack to Manage Support Cases
- Gestisci le risorse AWS nei canali Slack con Amazon Q Developer nelle applicazioni di chat

Servizi correlati:

- Amazon Q Developer nelle applicazioni di chat
- AWS Systems Manager Change Calendar
- AWS Systems Manager Documents

OPS03-BP05 Incoraggiamento alla sperimentazione

La sperimentazione è un catalizzatore per trasformare nuove idee in prodotti e funzionalità. La sperimentazione accelera l'apprendimento e mantiene acceso l'interesse e il coinvolgimento dei membri del team. I membri del team sono incoraggiati a sperimentare spesso per promuovere l'innovazione. Anche quando si verifica un risultato indesiderato, è comunque utile sapere quello che non bisogna fare. I membri del team non vengono puniti per gli esperimenti riusciti con risultati indesiderati.

Risultato desiderato:

- La tua organizzazione incoraggia la sperimentazione per promuovere l'innovazione.
- Gli esperimenti sono utilizzati come un'opportunità per imparare.

Anti-pattern comuni:

- Vuoi eseguire un test A/B, ma non esiste un meccanismo per eseguire l'esperimento. Distribuisci una modifica all'interfaccia utente senza la possibilità di testarla. Questo comporta un'esperienza cliente negativa.
- La tua azienda ha solo un ambiente di test e uno di produzione. Non esiste un ambiente di sperimentazione (sandbox) in cui provare nuove funzionalità o prodotti, per cui le sperimentazioni avvengono all'interno dell'ambiente di produzione.

Vantaggi dell'adozione di questa best practice:

- La sperimentazione incoraggia l'innovazione.
- Grazie alla sperimentazione puoi reagire più velocemente al feedback degli utenti.
- La tua organizzazione sviluppa una cultura dell'apprendimento.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Le sperimentazioni vanno eseguite in modo sicuro. Sfrutta più ambienti per sperimentare senza mettere a rischio le risorse di produzione. Usa il test A/B e le flag delle funzionalità per testare gli esperimenti. Offri ai membri del team la possibilità di eseguire esperimenti in un ambiente di sperimentazione (sandbox).

Esempio del cliente

AnyCompany Retail incoraggia la sperimentazione. I membri del team possono dedicare il 20% della propria settimana lavorativa alla sperimentazione o all'apprendimento di nuove tecnologie. Hanno a disposizione un ambiente di sperimentazione (sandbox) in cui possono innovare. Il test A/B viene utilizzato per nuove funzionalità che possono essere così convalidate con il feedback di utenti reali.

Passaggi dell'implementazione

1. Collabora con la direzione della tua organizzazione per supportare la sperimentazione. I membri del team devono essere incoraggiati a eseguire esperimenti in modo sicuro.
2. Offri ai membri del team un ambiente in cui possono sperimentare in modo sicuro (devono avere accesso a un ambiente simile alla produzione).
    a. Puoi usare un Account AWS separato per creare un ambiente sandbox e AWS Control Tower per allocare questi account.
3. Usa flag delle funzionalità e test A/B per sperimentare in modo sicuro e raccogliere il feedback degli utenti.
    a. La flag delle funzionalità AWS AppConfig consentono di creare flag delle funzionalità.
    b. Puoi utilizzare le versioni AWS Lambda per implementare una nuova versione di una funzione per il beta testing.

Livello di impegno per il piano di implementazione: elevato. Offrire ai membri del team un ambiente in cui sperimentare in modo sicuro può richiedere investimenti significativi. Potresti anche aver bisogno di modificare il codice dell'applicazione per usare flag di funzionalità o supportare il test A/B.

Risorse

Best practice correlate:

- OPS11-BP02 Eseguire l'analisi post-incidente: imparare dagli incidenti è un fattore importante di innovazione e sperimentazione.
- OPS11-BP03 Implementazione di circuiti di feedback: i cicli di feedback costituiscono una parte importante della sperimentazione.

Documenti correlati:

- An Inside Look at the Amazon Culture: Experimentation, Failure, and Customer Obsession
- Best practices for creating and managing sandbox accounts in AWS
- Create a Culture of Experimentation Enabled by the Cloud
- Enabling experimentation and innovation in the cloud at SulAmérica Seguros
- Experiment More, Fail Less
- Organizzazione dell'ambiente AWS che utilizza più account: unità organizzativa dell'ambiente di sperimentazione (sandbox).
- Using AWS AppConfig Feature Flags

Video correlati:

- AWS On Air ft. Amazon CloudWatch Evidently | AWS Events
- AWS On Air San Fran Summit 2022 ft. AWS AppConfig Feature Flags integration with Jira
- AWS re:Invent 2022 - A deployment is not a release: Control your launches w/feature flags (BOA305-R)
- Programmatically Create an Account AWS with AWS Control Tower
- Set Up a Multi-Account AWS Environment that Uses Best Practices for AWS Organizations

Esempi correlati:

- AWS Innovation Sandbox

- End-to-end Personalization 101 for E-Commerce

Servizi correlati:

- Amazon CloudWatch Evidently
- AWS AppConfig
- AWS Control Tower

OPS03-BP06 Incoraggiamento ai membri del team a mantenere e ampliare le proprie competenze

I team devono aumentare le proprie competenze per adottare nuove tecnologie e supportare i cambiamenti in termini di domanda e responsabilità a supporto dei carichi di lavoro. L'ampliamento delle competenze nelle nuove tecnologie è spesso fonte di soddisfazione per i membri del team e supporta l'innovazione. Incoraggia i membri del team a perseguire e mantenere le certificazioni di settore in modo da convalidare e riconoscere le loro crescenti competenze. Pratica la formazione trasversale per promuovere il trasferimento di conoscenze e ridurre il rischio di impatto significativo in caso di perdita di membri del team qualificati ed esperti con competenze a livello istituzionale. Fornisci tempo strutturato dedicato per la formazione.

AWS fornisce risorse, tra cui l'AWS Getting Started Resource Center, i blog AWS, i Tech Talk AWS online, eventi e webinar AWS e i AWS Well-Architected Labs, che forniscono indicazioni, esempi e procedure dettagliate per la formazione dei team.

Risorse come Supporto, (AWS re:Post, Supporto Center) e documentazione AWS rimuovono gli ostacoli tecnici e consentono di migliorare le operazioni. Se hai domande, contatta Supporto tramite Supporto Center.

AWS condivide anche best practice e modelli appresi attraverso la gestione di AWS nella Amazon Builders' Library, oltre a un'ampia varietà di ulteriore materiale didattico utile tramite il blog AWS e il podcast ufficiale AWS.

AWS Training and Certification offre formazione gratuita tramite corsi digitali personalizzati, oltre a corsi di formazione per ruolo o dominio. Per supportare ulteriormente lo sviluppo delle competenze AWS dei team, è anche possibile iscriversi a corsi di formazione con istruttore.

Risultato desiderato: la tua organizzazione valuta in modo costante le lacune nelle competenze e le colma con budget e investimenti strutturati. I team incoraggiano e incentivano i membri con attività di miglioramento delle competenze, come l'acquisizione delle principali certificazioni del settore.

I team traggono beneficio da programmi dedicati alla condivisione incrociata delle conoscenze, come corsi di formazione in pausa pranzo, giornate di full immersion, hackathon e giornate di gioco. L'organizzazione mantiene i sistemi delle conoscenze aggiornati e pertinenti per la formazione incrociata dei membri dei team, compresi i corsi di formazione per l'onboarding dei nuovi assunti.

Anti-pattern comuni:

- In assenza di un programma di formazione strutturato e di un budget, i team riscontrano difficoltà nel tentativo di tenere il passo con l'evoluzione della tecnologia, il che si traduce in un aumento dell'attrito.
- Nell'ambito della migrazione ad AWS, l'organizzazione dimostra lacune nelle competenze e una padronanza del cloud variabile tra i team. Senza un impegno per il miglioramento delle competenze, i team si ritrovano oberati di attività di gestione legacy e inefficienti dell'ambiente cloud, causando un aumento del lavoro degli operatori. Questo stato di esaurimento dei team aumenta l'insoddisfazione dei dipendenti.

Vantaggi dell'adozione di questa best practice: gli investimenti consapevoli della tua organizzazione nel miglioramento delle competenze dei propri team accelerano e scalano anche l'adozione e ottimizzazione del cloud. I programmi di formazione mirati favoriscono l'innovazione e creano capacità operative per consentire ai team di essere preparati a gestire gli eventi. I team investono consapevolmente nell'implementazione e nell'evoluzione delle best practice. Il morale dei team è alto e i membri apprezzano il contributo che offrono all'azienda.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Per adottare nuove tecnologie, promuovere l'innovazione e stare al passo con i cambiamenti in termini di domanda e responsabilità a supporto dei carichi di lavoro, investi continuamente nella crescita professionale dei team.

Passaggi dell’implementazione

1. Ricorri a programmi strutturati a sostegno del cloud: AWS Skills Guild offre formazione consultiva per aumentare la sicurezza nelle competenze cloud e promuovere una cultura della formazione continua.
2. Metti a disposizione le risorse per la formazione: metti a disposizione del tempo in modo strutturato e dedicato, accesso ai materiali di formazione, risorse di laboratorio e supporto alla partecipazione a conferenze e organizzazioni professionali che offrono opportunità di apprendimento da docenti

e colleghi. Offri ai membri dei team junior la possibilità di contattare i membri dei team senior affinché questi fungano da mentori o possano mostrare loro come lavorano trasmettendo metodi e competenze consolidati. Incoraggia l'apprendimento dei contenuti non direttamente correlati al lavoro per avere una prospettiva più ampia.

3. Incoraggia l'uso di risorse tecniche esperte: sfrutta risorse come AWS re:Post per accedere a conoscenze consolidate e a una vibrante community.
4. Crea e mantieni un repository di conoscenze aggiornato: utilizza piattaforme di condivisione delle conoscenze come wiki e runbook. Crea la tua fonte di conoscenza specialistica riutilizzabile con AWS re:Post Private per semplificare la collaborazione, migliorare la produttività e accelerare l'onboarding dei dipendenti.
5. Formazione del team e coinvolgimento tra team: pianifica le esigenze di formazione continua dei membri del tuo team. Offri loro l'opportunità di unirsi ad altri team (temporaneamente o definitivamente) per condividere competenze e best practice a beneficio dell'intera organizzazione.
6. Supporta il perseguimento e il mantenimento delle certificazioni di settore: favorisci l'acquisizione e il mantenimento da parte dei membri del tuo team di certificazioni di settore che convalidano quanto appreso e le loro conoscenze e riconoscono i loro risultati.

Livello di impegno per il piano di implementazione: elevato

Risorse

Best practice correlate:

- OPS03-BP01 Definizione della sponsorizzazione esecutiva
- OPS11-BP04 Gestione delle informazioni

Documenti correlati:

- Whitepaper AWS | Cloud Adoption Framework: People Perspective
- Investing in continuous learning to grow your organization's future
- AWS Skills Guild
- AWS Training and Certification
- Supporto
- AWS re:Post
- Centro risorse per le nozioni di base AWS

- Blog AWS
- Cloud AWS Conformità di
- AWS Documentazione di
- Il podcast ufficiale di AWS
- AWS Colloqui tecnici online su
- Eventi e webinar AWS
- AWS Well-Architected Labs
- Amazon Builders' Library

Video correlati:

- AWS re:Invent 2023 | Reskilling at the speed of cloud: Turning employees into entrepreneurs
- WS re:Invent 2023 | Building a culture of curiosity through gamification

OPS03-BP07 Team di risorse appropriati

Stabilisci il giusto numero di membri competenti del team e gli strumenti e le risorse per supportare le esigenze di carico di lavoro. Il sovraccarico dei membri del team aumenta il rischio di errore umano. Gli investimenti in strumenti e risorse, come l'automazione, consentono di scalare l'efficacia del team consentendogli di supportare un numero maggiore di carichi di lavoro senza richiedere capacità aggiuntiva.

Risultato desiderato:

- Hai assegnato al tuo team personale adeguato per acquisire le competenze necessarie a gestire i carichi di lavoro in conformità al tuo piano di migrazione. AWS Man mano che il team si è ampliato nel corso del progetto di migrazione, ha acquisito competenze nelle AWS tecnologie di base che l'azienda intende utilizzare per la migrazione o la modernizzazione delle applicazioni.
- Hai preparato con attenzione il piano per i membri del team per fare un uso efficiente delle risorse, sfruttando l'automazione e il flusso di lavoro. Un team più piccolo può ora gestire più infrastrutture per conto dei team di sviluppo delle applicazioni.
- Con il cambiamento delle priorità operative, qualsiasi vincolo di risorse viene identificato in modo proattivo per proteggere il successo delle iniziative aziendali.
- Le metriche che segnalano le difficoltà operative, ad esempio l'affaticamento da chiamata o il paging eccessivo, vengono esaminate per verificare che il personale non sia sovraccaricato.

Anti-pattern comuni:

- Il vostro personale non ha ancora migliorato AWS le proprie competenze man mano che state attuando il piano pluriennale di migrazione al cloud, il che rischia di sostenere i carichi di lavoro e di abbassare il morale dei dipendenti.
- L'intera organizzazione IT adotta le modalità di lavoro agili. L'azienda assegna le priorità al portafoglio di prodotti e stabilisce le metriche per le funzionalità che devono essere sviluppate per prime. Il processo agile non richiede che i team assegnino story point ai piani di lavoro. Di conseguenza, è impossibile conoscere il livello di capacità richiesto per il successivo lavoro o se le competenze giuste sono state assegnate al lavoro.
- Avete chiesto a un AWS partner di migrare i vostri carichi di lavoro e non disponete di un piano di transizione del supporto per i vostri team una volta che il partner avrà completato il progetto di migrazione. I team hanno difficoltà a supportare i carichi di lavoro in modo efficiente ed efficace.

Vantaggi dell'adozione di questa best practice: la tua organizzazione vanta membri del team con competenze adeguate a supportare i carichi di lavoro. L'allocazione delle risorse può adattarsi al cambiamento delle priorità senza influire sulle prestazioni. Il risultato è che i team sono in grado di supportare i carichi di lavoro, massimizzando al contempo il tempo per concentrarsi sull'innovazione per i clienti e aumentando a sua volta la soddisfazione dei dipendenti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

La pianificazione delle risorse per la migrazione al cloud deve avvenire a un livello organizzativo in linea con il piano di migrazione e l'implementazione del modello operativo desiderato per supportare il nuovo ambiente cloud. Ciò deve includere la comprensione delle tecnologie cloud utilizzate per i team di sviluppo aziendale e delle applicazioni. La leadership dell'infrastruttura e delle operazioni deve pianificare l'analisi del divario delle competenze, la formazione e la definizione dei ruoli per gli ingegneri che guidano l'adozione del cloud.

Passaggi dell'implementazione

1. Definisci i criteri per il successo dei team con metriche operative pertinenti, come la produttività del personale (ad esempio, i costi di supporto di un carico di lavoro o le ore spese dall'operatore per gli incidenti).

2. Definisci i meccanismi di pianificazione e ispezione della capacità delle risorse per verificare che il giusto equilibrio di capacità qualificata sia disponibile quando necessario e possa essere modificato nel tempo.
3. Crea meccanismi, ad esempio inviando un sondaggio mensile ai team, per comprendere le sfide legate al lavoro che hanno un impatto sui team, come l'aumento delle responsabilità, i cambiamenti nella tecnologia, la mancanza di personale o l'aumento dei clienti supportati.
4. Utilizza questi meccanismi per interagire con i team e individuare le tendenze che possono contribuire alle sfide relative alla produttività dei dipendenti. Quando i team sono influenzati da fattori esterni, rivaluta gli obiettivi e modifica i target in base alle esigenze. Individua gli ostacoli che impediscono i progressi dei team.
5. Verifica con regolarità se le risorse attualmente allocate sono ancora sufficienti o se occorre aggiungere e apportare le modifiche appropriate ai team di supporto.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS03-BP06 I membri del team sono incoraggiati a mantenere e accrescere le proprie competenze
- OPS09-BP03 Rivedi le metriche operative e dai priorità al miglioramento
- OPS10-BP01 Utilizza un processo per la gestione di eventi, incidenti e problemi
- OPS10-BP07 Automatizza le risposte agli eventi

Documenti correlati:

- Cloud AWS Framework di adozione: prospettiva delle persone
- Becoming a Future-Ready Enterprise
- Prioritize your Employees' Skills to Drive Business Growth
- Organizzazioni ad alte prestazioni: il team da due pizze Amazon
- How Cloud-Mature Enterprises Succeed

Preparazione

Questions

- OPS 4. Come si implementa l'osservabilità nel carico di lavoro?
- OPS 5. In che modo riduci i difetti, favorisci la correzione e migliori il flusso nella produzione?
- OPS 6. In che modo mitighi i rischi dell'implementazione?
- OPS 7. Come fai a sapere se hai tutto pronto per supportare un carico di lavoro?

OPS 4. Come si implementa l'osservabilità nel carico di lavoro?

Implementare l'osservabilità nel carico di lavoro ti permette di comprendere lo stato di quest'ultimo e di adottare decisioni basate sui dati e che riflettono i requisiti aziendali.

Best practice

- OPS04-BP01 Identifica gli indicatori chiave di prestazione
- OPS04-BP02 Implementare la telemetria delle applicazioni
- OPS04-BP03 Implementare la telemetria dell'esperienza utente
- OPS04-BP04 Implementazione della telemetria delle dipendenze
- OPS04-BP05 Implementare la tracciabilità distribuita

OPS04-BP01 Identifica gli indicatori chiave di prestazione

L'implementazione dell'osservabilità nel carico di lavoro inizia con la comprensione del suo stato e l'adozione di decisioni basate sui dati che riflettono i requisiti aziendali. Uno dei modi più efficaci per garantire l'allineamento tra le attività di monitoraggio e gli obiettivi aziendali consiste nella definizione e nel monitoraggio degli indicatori chiave di performance (). KPIs

Risultato desiderato: pratiche di osservabilità efficienti e strettamente allineate agli obiettivi aziendali garantiscono che le attività di monitoraggio siano sempre al servizio di risultati aziendali tangibili.

Anti-pattern comuni:

- IndefinitoKPIs: lavorare senza un sistema chiaro KPIs può portare a un monitoraggio eccessivo o insufficiente e alla mancanza di segnali vitali.
- StaticoKPIs: non rivisitare o perfezionare man mano che il carico di lavoro o KPIs gli obiettivi aziendali si evolvono.

- Disallineamento: concentrarsi su metriche tecniche non direttamente correlate ai risultati aziendali o che sono più difficili da correlare ai problemi del mondo reale.

Vantaggi dell'adozione di questa best practice:

- Facilità di identificazione dei problemi: le aziende KPIs spesso evidenziano i problemi in modo più chiaro rispetto alle metriche tecniche. Un calo aziendale KPI può individuare un problema in modo più efficace rispetto all'analisi di numerose metriche tecniche.
- Allineamento aziendale: assicura che le attività di monitoraggio supportino direttamente gli obiettivi aziendali.
- Efficienza: viene data la priorità alle risorse di monitoraggio e al focus sulle metriche che contano.
- Proattività: riconoscere e risolvere i problemi prima che abbiano implicazioni aziendali più ampie.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per definire in modo efficace il carico di lavoro: KPIs

1. Inizia con i risultati aziendali: prima di approfondire le metriche, comprendi i risultati aziendali desiderati. È stato rilevato un aumento delle vendite, un maggiore coinvolgimento degli utenti o tempi di risposta più rapidi?
2. Correla le metriche tecniche con gli obiettivi aziendali: non tutte le metriche tecniche influiscono direttamente sui risultati aziendali. Identifica quelli che lo fanno, ma spesso è più semplice identificare un problema utilizzando un'azienda. KPI
3. Usa Amazon CloudWatch: Employ CloudWatch per definire e monitorare le metriche che rappresentano le tue. KPIs
4. Rivedi e aggiorna regolarmenteKPIs: man mano che il carico di lavoro e la tua attività si evolvono, mantieni i tuoi dati pertinenti. KPIs
5. Coinvolgi le parti interessate: coinvolgi i team tecnici e aziendali nella definizione e nella revisione. KPIs

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- the section called “OPS04-BP02 Implementare la telemetria delle applicazioni”
- the section called “OPS04-BP03 Implementare la telemetria dell'esperienza utente”
- the section called “OPS04-BP04 Implementazione della telemetria delle dipendenze”
- the section called “OPS04-BP05 Implementare la tracciabilità distribuita”

Documenti correlati:

- AWS Migliori pratiche di osservabilità
- CloudWatch Guida per l'utente
- AWS Corso Observability Skill Builder

Video correlati:

- Developing an observability strategy

Esempi correlati:

- One Observability Workshop

OPS04-BP02 Implementare la telemetria delle applicazioni

La telemetria dell'applicazione è la base su cui si fonda l'osservabilità del carico di lavoro. È fondamentale emettere dati di telemetria che offrano approfondimenti utili sullo stato dell'applicazione e sul raggiungimento degli obiettivi sia tecnici sia aziendali. Dalla risoluzione dei problemi alla misurazione dell'impatto di una nuova funzionalità o alla garanzia dell'allineamento con gli indicatori chiave di prestazione aziendali (KPIs), la telemetria delle applicazioni influenza il modo in cui create, gestite ed evolvete il carico di lavoro.

Metriche, log e tracce costituiscono i tre pilastri principali dell'osservabilità. Questi operano come strumenti diagnostici che descrivono lo stato dell'applicazione. Nel tempo, aiutano a creare criteri di base e a identificare le anomalie. Tuttavia, per garantire l'allineamento tra le attività di monitoraggio e gli obiettivi aziendali, è fondamentale definire e monitorare. KPIs KPIsLe aziende spesso semplificano l'identificazione dei problemi rispetto alle sole metriche tecniche.

Altri tipi di telemetria, come il monitoraggio degli utenti in tempo reale (RUM) e le transazioni sintetiche, completano queste fonti di dati primarie. RUMoffre approfondimenti sulle interazioni degli

utenti in tempo reale, mentre le transazioni sintetiche simulano i potenziali comportamenti degli utenti, aiutando a individuare i colli di bottiglia prima che gli utenti reali li incontrino.

Risultato desiderato: ottieni approfondimenti utili sulle prestazioni del tuo carico di lavoro. Questi approfondimenti consentono di prendere decisioni proattive sull'ottimizzazione delle prestazioni, ottenere una maggiore stabilità del carico di lavoro, semplificare i processi CI/CD e utilizzare le risorse in modo efficace.

Anti-pattern comuni:

- Osservabilità incompleta: trascurare l'incorporazione dell'osservabilità a ogni livello del carico di lavoro, con conseguenti punti ciechi che possono nascondere le prestazioni vitali del sistema e gli approfondimenti sul comportamento.
- Visualizzazione frammentata dei dati: quando i dati sono sparsi su più strumenti e sistemi, diventa difficile mantenere una visione olistica dello stato e delle prestazioni del carico di lavoro.
- Problemi segnalati dagli utenti: un segno della mancanza di un rilevamento proattivo dei problemi tramite telemetria e monitoraggio aziendale. KPI

Vantaggi dell'adozione di questa best practice:

- Processo decisionale informato: con gli approfondimenti tratti dalla telemetria e dal business, puoi prendere decisioni basate sui dati. KPIs
- Migliore efficienza operativa: l'utilizzo delle risorse basato sui dati porta a un miglioramento dell'efficienza risparmiando sui costi.
- Maggiore stabilità del carico di lavoro: rilevamento e risoluzione più rapidi dei problemi con conseguente aumento dei tempi di attività.
- Processi CI/CD semplificati: gli approfondimenti ricavati dai dati di telemetria facilitano il perfezionamento dei processi e la distribuzione affidabile del codice.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per implementare la telemetria delle applicazioni per il tuo carico di lavoro, utilizza servizi AWS come Amazon e. CloudWatch AWS X-Ray Amazon CloudWatch offre una suite completa di strumenti di monitoraggio che ti consentono di osservare le tue risorse e applicazioni in ambienti locali AWS e locali. Raccoglie, tiene traccia e analizza le metriche, consolida e monitora i dati di log e risponde

alle modifiche che interessano le risorse, migliorando la comprensione del funzionamento del carico di lavoro. In parallelo, ti AWS X-Ray consente di tracciare, analizzare ed eseguire il debug delle tue applicazioni, offrendoti una comprensione approfondita del comportamento del tuo carico di lavoro. Grazie a funzionalità come mappe dei servizi, distribuzioni della latenza e tempistiche di tracciamento, AWS X-Ray fornisce informazioni dettagliate sulle prestazioni del carico di lavoro e sui colli di bottiglia che lo influiscono.

Passaggi dell'implementazione

1. Identifica quali dati raccogliere: definisci le metriche, i log e le tracce essenziali che potrebbero offrire importanti informazioni dettagliate sullo stato, le prestazioni e il comportamento del tuo carico di lavoro.
2. Implementa l'CloudWatchagente: l' CloudWatch agente è fondamentale nell'acquisizione dei parametri e dei log di sistema e delle applicazioni dal carico di lavoro e dall'infrastruttura sottostante. L' CloudWatch agente può essere utilizzato anche per raccogliere OpenTelemetry o inviare tracce a raggi X e inviarle a X-Ray.
3. Implementa il rilevamento delle anomalie per log e metriche: utilizza il rilevamento delle anomalie CloudWatch nei log e il rilevamento delle anomalie nelle CloudWatchmetriche per identificare automaticamente le attività insolite nelle operazioni dell'applicazione. Questi strumenti utilizzano algoritmi di machine learning per rilevare e comunicare le anomalie, migliorando le capacità di monitoraggio e accelerando i tempi di risposta a potenziali interruzioni o minacce alla sicurezza. Configura queste funzionalità per gestire in modo proattivo lo stato e la sicurezza delle applicazioni.
4. Proteggi i dati sensibili dei log: utilizza la protezione dei dati di Amazon CloudWatch Logs per mascherare le informazioni sensibili all'interno dei tuoi log. Questa funzionalità aiuta a mantenere la privacy e la conformità con il rilevamento e il mascheramento automatici dei dati sensibili prima dell'accesso. Implementa il mascheramento dei dati per gestire e proteggere in modo sicuro i dettagli sensibili come le informazioni di identificazione personale (). PII
5. Definisci e monitora il businessKPIs: stabilisci metriche personalizzate in linea con i risultati aziendali.
6. Strumenta la tua applicazione con AWS X-Ray: oltre a implementare l' CloudWatchagente, è fondamentale strumentare l'applicazione per emettere dati di traccia. Questo processo può fornire ulteriori approfondimenti sul comportamento e sulle prestazioni del carico di lavoro.
7. Standardizza la raccolta dei dati nell'applicazione: standardizza le pratiche di raccolta dei dati nell'intera applicazione. L'uniformità aiuta a correlare e analizzare i dati, fornendo una visione completa del comportamento dell'applicazione.

8. Implementa l'osservabilità tra account: migliora l'efficienza del monitoraggio su più account con l'osservabilità tra più account di Account AWS Amazon CloudWatch . Con questa funzionalità, puoi consolidare metriche, log e allarmi di diversi account in un'unica visualizzazione, semplificando la gestione e migliorando i tempi di risposta per i problemi identificati nell'ambiente dell'organizzazione. AWS
9. Analizza e agisci in base ai dati: una volta completata la raccolta e la normalizzazione dei dati, usa Amazon CloudWatch per l'analisi di metriche e log e AWS X-Rayper l'analisi delle tracce. Tale analisi può fornire approfondimenti cruciali sullo stato, le prestazioni e il comportamento del carico di lavoro, guidando il processo decisionale.

Livello di impegno per il piano di implementazione: elevato

Risorse

Best practice correlate:

- OPS04-BP01 Definisci il carico di lavoro KPIs
- OPS04-BP03 Implementare la telemetria delle attività degli utenti
- OPS04-BP04 Implementare la telemetria delle dipendenze
- OPS04-BP05 Implementare la tracciabilità delle transazioni

Documenti correlati:

- AWS Observability Best Practices
- Guida per l'utente di CloudWatch
- AWS X-Ray Guida per gli sviluppatori
- Strumentazione di sistemi distribuiti per visibilità operativa
- AWS Observability Skill Builder Course
- Cosa c'è di nuovo con Amazon CloudWatch
- Cosa c'è di nuovo con AWS X-Ray

Video correlati:

- AWS re:Invent 2022 - Le migliori pratiche di osservabilità su Amazon
- AWS re:Invent 2022 - Sviluppo di una strategia di osservabilità

Esempi correlati:

- One Observability Workshop
- AWS Libreria di soluzioni: monitoraggio delle applicazioni con Amazon CloudWatch

OPS04-BP03 Implementare la telemetria dell'esperienza utente

Acquisire informazioni approfondite sulle esperienze dei clienti e sulle interazioni con la tua applicazione è fondamentale. Il monitoraggio degli utenti reali (RUM) e le transazioni sintetiche sono strumenti potenti per questo scopo. RUMfornisce dati sulle interazioni reali degli utenti garantendo una prospettiva non filtrata della soddisfazione degli utenti, mentre le transazioni sintetiche simulano le interazioni degli utenti, aiutando a rilevare potenziali problemi ancor prima che abbiano un impatto sugli utenti reali.

Risultato desiderato: una visione olistica dell'esperienza del cliente, il rilevamento proattivo dei problemi e l'ottimizzazione delle interazioni degli utenti per offrire esperienze digitali fluide.

Anti-pattern comuni:

- Applicazioni senza monitoraggio reale degli utenti (): RUM
  - Rilevamento ritardato dei problemi: in caso contrarioRUM, potreste non accorgervi dei rallentamenti o dei problemi di prestazioni fino a quando gli utenti non si lamentano. Questo approccio reattivo può causare insoddisfazione nei clienti.
  - Mancanza di informazioni sull'esperienza utente: non utilizzarla RUM significa perdere dati cruciali che mostrano come gli utenti reali interagiscono con l'applicazione, limitando la capacità di ottimizzare l'esperienza utente.
- Applicazioni senza transazioni sintetiche:
  - Casi limite trascurati: le transazioni sintetiche consentono di testare percorsi e funzioni che potrebbero non essere utilizzati frequentemente dagli utenti tipici, ma che sono fondamentali per determinate funzioni aziendali. Senza di esse, questi percorsi potrebbero non funzionare correttamente e passare inosservati.
  - Verifica della presenza di problemi quando l'applicazione non viene utilizzata: i test sintetici regolari possono simulare situazioni in cui gli utenti reali non interagiscono attivamente con l'applicazione, garantendo che il sistema funzioni sempre correttamente.

Vantaggi dell'adozione di questa best practice:

- Rilevamento proattivo dei problemi: identifica e risolvi i problemi potenziali prima che abbiano un impatto sugli utenti reali.
- Esperienza utente ottimizzata: il feedback continuo fornito RUM aiuta a perfezionare e migliorare l'esperienza utente complessiva.
- Informazioni approfondite sulle prestazioni del dispositivo e del browser: scopri come si comporta la tua applicazione in vari dispositivi e browser e implementa ulteriori ottimizzazioni.
- Flussi di lavoro aziendali convalidati: transazioni sintetiche regolari assicurano che le funzionalità principali e i percorsi critici siano operativi ed efficienti in maniera costante.
- Prestazioni delle applicazioni migliorate: sfrutta le informazioni approfondite raccolte dai dati degli utenti reali per migliorare la reattività e l'affidabilità delle applicazioni.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per sfruttare RUM e sintetizzare le transazioni per la telemetria delle attività degli utenti, offre AWS servizi come Amazon e Amazon CloudWatch RUM Synthetics. CloudWatch Metriche, log e tracce, insieme ai dati sulle attività degli utenti, forniscono una visione completa dello stato operativo dell'applicazione e dell'esperienza utente.

Passaggi dell'implementazione

1. Implementa Amazon CloudWatch RUM: integra la tua applicazione con CloudWatch RUM per raccogliere, analizzare e presentare dati utente reali.
   a. Usa la CloudWatch RUM JavaScript libreria per l'integrazione RUM con la tua applicazione.
   b. Configura pannelli di controllo per visualizzare e monitorare i dati relativi agli utenti reali.
2. Configura CloudWatch Synthetics: crea canaries, o routine con script, che simulano le interazioni degli utenti con la tua applicazione.
   a. Definisci i flussi di lavoro e i percorsi critici delle applicazioni.
   b. Progetta canarini utilizzando gli script CloudWatch Synthetics per simulare le interazioni degli utenti per questi percorsi.
   c. Pianifica e monitora i canary affinché si attivino a intervalli specifici, in modo da garantire controlli costanti delle prestazioni.
3. Analizza e agisci in base ai dati: utilizza i dati e le transazioni sintetiche per ottenere informazioni RUM e adottare misure correttive quando vengono rilevate anomalie. Utilizza CloudWatch dashboard e allarmi per rimanere informato.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS04-BP01 Identifica gli indicatori chiave di prestazione
- OPS04-BP02 Implementare la telemetria delle applicazioni
- OPS04-BP04 Implementazione della telemetria delle dipendenze
- OPS04-BP05 Implementare la tracciabilità distribuita

Documenti correlati:

- CloudWatch RUMGuida Amazon
- Guida Amazon CloudWatch Synthetics

Video correlati:

- Ottimizza le applicazioni attraverso approfondimenti sugli utenti finali con Amazon CloudWatch RUM
- AWS su Air ft. Monitoraggio degli utenti in tempo reale per Amazon CloudWatch

Esempi correlati:

- One Observability Workshop
- Repository Git per Amazon CloudWatch RUM Web Client
- Utilizzo di Amazon CloudWatch Synthetics per misurare il tempo di caricamento delle pagine

OPS04-BP04 Implementazione della telemetria delle dipendenze

La telemetria delle dipendenze è essenziale per monitorare lo stato e le prestazioni dei servizi e dei componenti esterni su cui si basa il carico di lavoro. Fornisce preziosi approfondimenti su reperibilità, timeout e altri eventi critici correlati alle dipendenze come DNS, database o API di terze parti. Dotando l'applicazione di strumenti per generare metriche, log e tracce relative a queste dipendenze, acquisisci una comprensione più chiara dei potenziali colli di bottiglia, problemi di prestazioni o errori che potrebbero influire sul carico di lavoro.

Risultato desiderato: le dipendenze su cui si basa il carico di lavoro funzionano come previsto, consentendo di gestire i problemi in modo proattivo e garantendo prestazioni ottimali del carico di lavoro.

Anti-pattern comuni:

- Scarsa attenzione alle dipendenze esterne: il focus è rivolto esclusivamente alle metriche interne dell'applicazione, trascurando quelle legate alle dipendenze esterne.
- Mancanza di monitoraggio proattivo: si attende che si verifichino problemi anziché monitorare costantemente lo stato e le prestazioni delle dipendenze.
- Monitoraggio isolato in comparti: utilizzo di strumenti di monitoraggio multipli ed eterogenei che possono portare a visioni dello stato delle dipendenze frammentate e incoerenti.

Vantaggi dell'adozione di questa best practice:

- Maggiore affidabilità del carico di lavoro: viene garantito che le dipendenze esterne siano costantemente disponibili e funzionino in modo ottimale.
- Rilevamento e risoluzione dei problemi più rapidi: identificazione e risoluzione proattiva dei problemi relativi alle dipendenze prima che influiscano sul carico di lavoro.
- Visione completa: acquisizione di una visione olistica dei componenti interni ed esterni che influenzano lo stato del carico di lavoro.
- Scalabilità del carico di lavoro migliorata: grazie alla comprensione dei limiti di scalabilità e delle caratteristiche prestazionali delle dipendenze esterne.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Implementa la telemetria delle dipendenze iniziando con l'identificazione dei servizi, dell'infrastruttura e dei processi da cui dipende il carico di lavoro. Esegui una valutazione quantitativa delle condizioni ottimali nelle quali tali dipendenze funzionano come previsto e poi determina quali dati sono necessari per misurarle. Con queste informazioni, puoi creare dashboard e avvisi che forniscono approfondimenti ai tuoi team operativi sullo stato di tali dipendenze. Usa gli strumenti AWS per scoprire e quantificare gli impatti quando le dipendenze non riescono a fornire le prestazioni necessarie. Riesamina costantemente la tua strategia per tenere conto dei cambiamenti relativi a priorità, obiettivi e alle informazioni dettagliate acquisite.

Passaggi dell'implementazione

Per implementare efficacemente la telemetria delle dipendenze:

1. Identifica le dipendenze esterne: collabora con le parti interessate per individuare le dipendenze esterne sulle quali si basa il tuo carico di lavoro. Le dipendenze esterne possono comprendere servizi come database esterni, API di terze parti, percorsi di connettività di rete verso altri ambienti e servizi DNS. Il primo passo verso un'efficace telemetria delle dipendenze è acquisire una comprensione totale di quali esse siano.
2. Sviluppa una strategia di monitoraggio: una volta acquisito un quadro chiaro delle dipendenze esterne, progetta una strategia di monitoraggio ad hoc per esse. Trovare la strategia giusta implica comprendere le criticità di tutte le dipendenze, il loro comportamento previsto e gli eventuali accordi od obiettivi sul livello di servizio associato (SLA o SLT). Imposta avvisi proattivi che ti informino riguardo a cambiamenti di stato o deviazioni delle prestazioni.
3. Usa il monitoraggio della rete: utilizza Internet Monitor e Network Monitor per informazioni complete sulle condizioni globali di Internet e della rete. Questi strumenti consentono di comprendere e rispondere alle interruzioni, ai malfunzionamenti o al degrado delle prestazioni che influiscono sulle dipendenze esterne.
4. Resta aggiornato con AWS Health: AWS Health è la fonte autorevole di informazioni sull'integrità delle risorse Cloud AWS. Utilizza AWS Health per visualizzare e ricevere notifiche su eventuali eventi di servizio in corso e modifiche imminenti, come gli eventi pianificati del ciclo di vita, in modo da poter adottare misure per mitigare gli impatti.
   a. Crea notifiche di eventi AWS Health personalizzati per i canali e-mail e chat con Notifiche all'utente AWS e integra a livello di codice con gli strumenti di monitoraggio e avviso di Amazon EventBridge o l'AWS Health API.
   b. Pianifica e monitora i progressi relativi agli eventi sull'integrità che richiedono un'azione integrando con strumenti di gestione delle modifiche o ITSM (come Jira ServiceNow) che potresti già utilizzare tramite Amazon EventBridge o l'API AWS Health.
   c. Se utilizzi AWS Organizations, abilita la visualizzazione dell'organizzazione per AWS Health per aggregare gli eventi AWS Health tra gli account.
5. Dota la tua applicazione di strumenti con AWS X-Ray: AWS X-Ray fornisce informazioni dettagliate sulle prestazioni delle applicazioni e delle relative dipendenze sottostanti. La tracciatura delle richieste dall'inizio alla fine ti permette di identificare colli di bottiglia o guasti nei servizi o nei componenti esterni su cui si basa l'applicazione.
6. Usa Amazon DevOps Guru: questo servizio basato sul machine learning identifica i problemi operativi, prevede quando potrebbero verificarsi problemi critici e consiglia azioni specifiche da

intraprendere. Fornisce un supporto prezioso per acquisire approfondimenti sulle dipendenze e assicurarsi che queste non siano la fonte di problemi operativi.

7. Monitora regolarmente: monitora le metriche e i log relativi alle dipendenze esterne in maniera costante. Imposta avvisi per comportamenti imprevisti o prestazioni ridotte.
8. Convalida dopo le modifiche: ogni volta che una dipendenza esterna è interessata da un aggiornamento o una modifica, convalidane le prestazioni e verifica che queste siano in linea con i requisiti dell'applicazione.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS04-BP02 Implementazione della telemetria dell'applicazione
- OPS04-BP03 Implementazione della telemetria dell'attività degli utenti
- OPS04-BP05 Implementazione della tracciabilità delle transazioni
- OPS08-BP04 Creare avvisi fruibili

Documenti correlati:

- Guida per l'utente delle Health Dashboard Amazon Personal
- AWS Internet Monitor User Guide
- Guida per sviluppatori di AWS X-Ray
- Guida per l'utente di AWS DevOps Guru

Video correlati:

- Visibility into how internet issues impact app performance
- Introduction to Amazon DevOps Guru
- Manage resource lifecycle events at scale with AWS Health

Esempi correlati:

- AWS Health Aware
- Using Tag-Based Filtering to Manage AWS Health Monitoring and Alerting at Scale

OPS04-BP05 Implementare la tracciabilità distribuita

Il tracciamento distribuito offre un modo per monitorare e visualizzare le richieste mentre attraversano vari componenti di un sistema distribuito. Acquisendo i dati di tracciamento da più fonti e analizzandoli in una vista unificata, i team possono comprendere meglio il flusso delle richieste, in quali punti sono presenti colli di bottiglia e dove devono concentrare gli sforzi di ottimizzazione.

Risultato desiderato: una visione olistica del flusso delle richieste nel tuo sistema distribuito, che ti permette di ottenere un debug preciso, prestazioni ottimizzate e migliori esperienze utente.

Anti-pattern comuni:

- Strumentazione incoerente: non tutti i servizi in un sistema distribuito sono dotati di strumentazione per il monitoraggio.
- Ignorare la latenza: concentrarsi solo sugli errori e non considerare la latenza o il graduale deterioramento delle prestazioni.

Vantaggi dell'adozione di questa best practice:

- Panoramica completa del sistema: visualizzazione dell'intero percorso delle richieste, dall'ingresso all'uscita.
- Debug avanzato: identificazione rapida dei punti in cui si verificano guasti o problemi di prestazioni.
- Esperienza utente migliorata: monitoraggio e ottimizzazione in base ai dati effettivi dell'utente, garantendo che il sistema soddisfi le esigenze del mondo reale.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Inizia identificando tutti gli elementi del carico di lavoro che richiedono strumentazione. Una volta presi in considerazione tutti i componenti, sfrutta strumenti come AWS X-Ray e OpenTelemetry per raccogliere dati di traccia per l'analisi con strumenti come X-Ray e Amazon Map. CloudWatch ServiceLens Partecipa a revisioni periodiche con gli sviluppatori e integra queste discussioni con strumenti come Amazon DevOps Guru, X-Ray Analytics e X-Ray Insights per aiutarti a scoprire

risultati più approfonditi. Imposta avvisi basati sui dati di tracciamento per notificare quando i risultati sono a rischio, come definito nel piano di monitoraggio del carico di lavoro.

Passaggi dell'implementazione

Per implementare il tracciamento distribuito in modo efficace:

1. Adotta AWS X-Ray: implementa X-Ray nella tua applicazione per ottenere informazioni dettagliate sul suo comportamento, comprenderne le prestazioni e individuare i punti critici. Utilizza X-Ray Insights per l'analisi automatica dei tracciamenti.
2. Strumenta i tuoi servizi: verifica che ogni servizio, da una AWS Lambdafunzione a un'EC2istanza, invii dati di traccia. Maggiore è il numero di servizi che offri, più chiara è la end-to-end visione.
3. Incorpora il monitoraggio degli utenti CloudWatch reali e il monitoraggio sintetico: integra il monitoraggio degli utenti reali (RUM) e il monitoraggio sintetico con X-Ray. Ciò ti consente di acquisire esperienze utenti del mondo reale e simulare le interazioni degli utenti per identificare potenziali problemi.
4. Usa l'CloudWatch agente: l'agente può inviare tracce da raggi X o OpenTelemetry, migliorando la profondità delle informazioni ottenute.
5. Usa Amazon DevOps Guru: DevOps Guru utilizza i dati di X-Ray, CloudWatch AWS Config, e AWS CloudTrail per fornire consigli pratici.
6. Analizza le tracce: esamina regolarmente i dati di tracciamento per individuare schemi, anomalie o colli di bottiglia che possono influire sulle prestazioni dell'applicazione.
7. Imposta avvisi: configura gli allarmi per schemi insoliti o latenze prolungate, CloudWatchper una risoluzione proattiva dei problemi.
8. Miglioramento continuo: riesamina la tua strategia di tracciamento man mano che aggiungi o modifichi servizi per acquisire tutti i punti dati pertinenti.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS04-BP01 Identifica gli indicatori chiave di prestazione
- OPS04-BP02 Implementare la telemetria delle applicazioni
- OPS04-BP03 Implementare la telemetria dell'esperienza utente

- OPS04-BP04 Implementazione della telemetria delle dipendenze

Documenti correlati:

- AWS X-Ray Guida per gli sviluppatori
- Guida per CloudWatch l'utente dell'agente Amazon
- Guida per l'utente di Amazon DevOps Guru

Video correlati:

- Usa Insights AWS X-Ray
- AWS su Air ft. Osservabilità: Amazon CloudWatch e AWS X-Ray

Esempi correlati:

- Strumentazione della tua applicazione per AWS X-Ray

OPS 5. In che modo riduci i difetti, favorisci la correzione e migliori il flusso nella produzione?

Adotta approcci che migliorino il flusso delle modifiche nella produzione, che attivino la rifattorizzazione e il feedback veloce su qualità e correzione di errori. Tali approcci accelerano l'ingresso in produzione delle modifiche vantaggiose, contengono i problemi che si sono diffusi e permettono di ottenere una rapida identificazione e risoluzione dei problemi introdotti attraverso le attività di implementazione.

Best practice

- OPS05-BP01 Utilizzo del controllo delle versioni
- OPS05-BP02 Test e convalida delle modifiche
- OPS05-BP03 Utilizzo di sistemi di gestione delle configurazioni
- OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione
- OPS05-BP05 Esecuzione della gestione delle patch
- OPS05-BP06 Condividi gli standard di progettazione
- OPS05-BP07 Implementazione di prassi per migliorare la qualità del codice

- OPS05-BP08 Utilizzo di più ambienti
- OPS05-BP09 Apporta modifiche frequenti, piccole e reversibili
- OPS05-BP10 Automazione completa dell'integrazione e dell'implementazione

OPS05-BP01 Utilizzo del controllo delle versioni

Utilizza il controllo delle versioni per attivare il monitoraggio di modifiche e rilasci.

Molti servizi AWS offrono funzionalità di controllo delle versioni. Utilizza un sistema di revisione o di controllo del codice sorgente, come ad esempio Git, per la gestione di codice e altri artefatti, come i modelli AWS CloudFormation con controllo delle versioni della tua infrastruttura.

Risultato desiderato: collaborazione dei team nell'ambito del codice. Una volta unito, il codice è coerente e nessuna modifica viene persa. Gli errori possono essere facilmente ripristinati mediante il corretto controllo delle versioni.

Anti-pattern comuni:

- Hai sviluppato e archiviato il codice sulla workstation. Si è verificato un errore di archiviazione non recuperabile sulla workstation e il codice è andato perso.
- Dopo aver sovrascritto il codice esistente con le modifiche, riavvii l'applicazione e non è più utilizzabile. Non è possibile ripristinare la modifica.
- Hai un blocco di scrittura su un file di report che deve essere modificato da altri utenti. Ti contattano per chiederti di smettere di utilizzarlo in modo che possano completare le loro attività.
- Il team di ricerca ha lavorato a un'analisi dettagliata che definisce il tuo lavoro futuro. Qualcuno ha salvato accidentalmente la lista della spesa nel report finale. Non puoi ripristinare la modifica e devi ricreare il report.

Vantaggi dell'adozione di questa best practice: grazie alle funzionalità di controllo delle versioni, puoi ripristinare facilmente gli stati validi noti e le versioni precedenti e limitare il rischio di perdita degli asset.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Mantieni gli asset in repository con controllo delle versioni. In questo modo si supporta il monitoraggio delle modifiche, l'implementazione di nuove versioni, il rilevamento delle modifiche apportate alle

versioni esistenti e il ripristino delle versioni precedenti, ad esempio il rollback a uno stato corretto noto in caso di errore. Integra nelle tue procedure le funzionalità di controllo delle versioni dei sistemi di gestione delle configurazioni.

Risorse

Best practice correlate:

- OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione

Video correlati:

- AWS re:Invent 2.023 - How Lockheed Martin builds software faster, powered by DevSecOps
- AWS re:Invent 2.023 - How GitHub operationalizes AI for team collaboration and productivity

OPS05-BP02 Test e convalida delle modifiche

Ogni modifica apportata deve essere testata per evitare errori in produzione. Questa best practice si concentra sulla verifica delle modifiche dal controllo di versione alla creazione dell'artefatto. Oltre alle modifiche al codice dell'applicazione, i test dovrebbero includere l'infrastruttura, la configurazione, i controlli di sicurezza e le procedure operative. I test assumono molte forme, dai test di unità all'analisi dei componenti software (SCA). Spostando i test più a sinistra nel processo di integrazione e consegna del software ottieni una maggiore certezza della qualità degli artefatti.

L'organizzazione deve sviluppare standard di test per tutti gli artefatti software. I test automatizzati riducono la fatica ed evitano gli errori dei test manuali. I test manuali potrebbero essere necessari in alcuni casi. Gli sviluppatori devono avere accesso ai risultati dei test automatizzati per creare cicli di feedback che migliorino la qualità del software.

Risultato desiderato: le modifiche software vengono testate prima del rilascio. Gli sviluppatori hanno accesso ai risultati dei test e alle convalide. La tua organizzazione ha uno standard per i test che applica a tutte le modifiche software.

Anti-pattern comuni:

- Implementi una nuova modifica software senza test. Non funziona in produzione e genera un'interruzione.
- I nuovi gruppi di sicurezza vengono implementati con AWS CloudFormation senza essere testati in un ambiente di pre-produzione. I gruppi di sicurezza rendono la tua app irraggiungibile per i clienti.

- Un metodo viene modificato, ma non ci sono test di unità. Il software ha esito negativo quando viene implementato in produzione.

Vantaggi dell'adozione di questa best practice: riduzione della percentuale di errori di modifica delle implementazioni software. La qualità del software viene migliorata. Gli sviluppatori hanno una maggiore consapevolezza della fattibilità del loro codice. Le policy di sicurezza possono essere implementate in maniera affidabile per supportare la conformità dell'organizzazione. Le modifiche all'infrastruttura, come gli aggiornamenti automatici delle policy di dimensionamento, vengono testate in anticipo per soddisfare le esigenze del traffico.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I test vengono eseguiti su tutte le modifiche, dal codice dell'applicazione all'infrastruttura, come parte della pratica di integrazione continua. I risultati dei test vengono pubblicati in modo che gli sviluppatori abbiano un feedback rapido. La tua organizzazione ha uno standard per i test che applica a tutte le modifiche software.

Usa la potenza dell'IA generativa con Amazon Q Developer per migliorare la produttività degli sviluppatori e la qualità del codice. Amazon Q Developer include la generazione di suggerimenti di codice (basati su modelli linguistici di grandi dimensioni), la produzione di test di unità (comprese le condizioni limite) e il miglioramento della sicurezza del codice tramite il rilevamento e la correzione delle vulnerabilità di sicurezza.

Esempio del cliente

Nell'ambito della pipeline di integrazione continua, AnyCompany Retail esegue diversi tipi di test su tutti gli artefatti software. L'azienda lo sviluppo guidato dai test, per cui tutto il software è dotato di test di unità. Una volta creato l'artefatto, eseguono test end-to-end. Al termine di questa prima serie di test, viene eseguita una scansione statica della sicurezza dell'applicazione, alla ricerca di vulnerabilità note. Gli sviluppatori ricevono messaggi al superamento di ciascun gate di test. Una volta completati tutti i test, l'artefatto software viene archiviato in un repository di artefatti.

Passaggi dell'implementazione

1. Collaborare con le parti interessate dell'organizzazione per sviluppare uno standard di test per gli artefatti software. Quali test standard devono superare tutti gli artefatti? Ci sono requisiti di conformità o di governance che devono essere inclusi nella copertura dei test? Devi condurre test di qualità del codice? Quando i test sono terminati, chi deve esserne a conoscenza?

1. La AWS Deployment Pipeline Reference Architecture contiene un elenco autorevole dei tipi di test che possono essere condotti su artefatti software come parte di una pipeline di integrazione.

2. Dota la tua applicazione di strumenti con i test necessari in base allo standard di test del software. Ogni set di test deve essere completato in meno di dieci minuti. I test devono essere eseguiti come parte della pipeline di integrazione.
   a. Usa Amazon Q Developer, uno strumento di IA generativa utile per creare casi di test di unità (comprese le condizioni limite), generare funzioni utilizzando codice e commenti e implementare algoritmi noti.
   b. Usa il revisore Amazon CodeGuru per testare il codice dell'applicazione in cerca di eventuali difetti.
   c. Puoi usare per AWS CodeBuild condurre i test su artefatti software.
   d. AWS CodePipeline può orchestrare i test software in una pipeline.

Risorse

Best practice correlate:

- OPS05-BP01 Utilizzo del controllo delle versioni
- OPS05-BP06 Condivisione degli standard di progettazione
- OPS05-BP07 Implementazione di prassi per migliorare la qualità del codice
- OPS05-BP10 Automazione completa dell'integrazione e dell'implementazione

Documenti correlati:

- Adozione di un approccio di sviluppo basato su test
- Accelerate your Software Development Lifecycle with Amazon Q
- Amazon Q Developer, now generally available, includes previews of new capabilities to reimagine developer experience
- The Ultimate Cheat Sheet for Using Amazon Q Developer in Your IDE
- Shift-Left Workload, leveraging AI for Test Creation
- Amazon Q Developer Center
- 10 ways to build applications faster with Amazon CodeWhisperer
- Looking beyond code coverage with Amazon CodeWhisperer

- Best Practices for Prompt Engineering with Amazon CodeWhisperer
- Automated AWS CloudFormation Testing Pipeline with TaskCat and CodePipeline
- Building end-to-end AWS DevSecOps CI/CD pipeline with open source SCA, SAST, and DAST tools
- Getting started with testing serverless applications
- My CI/CD pipeline is my release captain
- Practicing Continuous Integration and Continuous Delivery on AWS Whitepaper

Video correlati:

- Implement an API with Amazon Q Developer Agent for Software Development
- Installing, Configuring, & Using Amazon Q Developer with JetBrains IDEs (How-to)
- Mastering the art of Amazon CodeWhisperer - YouTube playlist
- AWS re:Invent 2020: Testable infrastructure: Integration testing on AWS
- AWS Summit ANZ 2021 - Driving a test-first strategy with CDK and test driven development
- Testing Your Infrastructure as Code with AWS CDK

Risorse correlate:

- AWS Deployment Pipeline Reference Architecture - Application
- AWS Kubernetes DevSecOps Pipeline
- Run unit tests for a Node.js application from GitHub by using AWS CodeBuild
- Use Serverspec for test-driven development of infrastructure code

Servizi correlati:

- Amazon Q Developer
- Revisore Amazon CodeGuru
- AWS CodeBuild
- AWS CodePipeline

OPS05-BP03 Utilizzo di sistemi di gestione delle configurazioni

L'utilizzo di sistemi di gestione delle configurazioni permette di effettuare modifiche alle stesse e tenerne traccia. Questi sistemi riducono gli errori causati dai processi manuali e il livello di impegno richiesto per la distribuzione delle modifiche.

Durante l'inizializzazione di una risorsa, la gestione delle configurazioni statiche consente di impostare valori che dovrebbero rimanere coerenti per tutta la vita utile della risorsa. Al momento dell'inizializzazione, la gestione delle configurazioni dinamiche consente di impostare valori che possono cambiare nel corso della vita utile di una risorsa. Ad esempio, è possibile impostare un interruttore per la funzionalità in modo da attivarla nel codice tramite una modifica della configurazione o modificare il livello di dettaglio del log durante un incidente.

Le configurazioni vanno implementate in uno stato noto e coerente. Utilizza l'ispezione automatizzata per monitorare in modo continuo le configurazioni delle risorse tra ambienti e regioni. Occorre definire questi controlli come codice e gestione automatizzati per garantire l'applicazione coerente delle regole in tutti gli ambienti. Le modifiche alle configurazioni vanno aggiornate tramite procedure di controllo delle modifiche concordate e applicate in modo coerente, rispettando il controllo delle versioni. Occorre gestire la configurazione dell'applicazione in modo indipendente rispetto al codice dell'applicazione e all'infrastruttura. In questo modo, si garantisce un'implementazione coerente tra più ambienti. Le modifiche alla configurazione non comportano la ricostruzione o la nuova implementazione dell'applicazione.

Risultato desiderato: puoi configurare, convalidare e implementare come parte della tua pipeline di integrazione continua e di distribuzione continua (CI/CD). Esegui il monitoraggio per verificare che le configurazioni siano corrette. Ciò riduce al minimo l'impatto sugli utenti finali e sui clienti.

Anti-pattern comuni:

- Aggiorni manualmente la configurazione del server Web all'interno del parco istanze e un certo numero di server non risponde a causa di errori di aggiornamento.
- Aggiorni manualmente il parco istanze del server applicazioni nel corso di molte ore. L'incoerenza nella configurazione durante la modifica causa comportamenti imprevisti.
- Qualcuno ha aggiornato i tuoi gruppi di sicurezza e i server Web non sono più accessibili. Senza sapere cosa è stato modificato, dedichi molto tempo a esaminare il problema prolungando il tempo necessario per il ripristino.
- Avvii una configurazione di preproduzione in produzione tramite CI/CD senza una convalida. Esponi utenti e clienti a dati e servizi errati.

Vantaggi dell'adozione di questa best practice: l'adozione di sistemi di gestione della configurazione riduce il livello di impegno necessario per apportare e tenere traccia delle modifiche e la frequenza degli errori causati dalle procedure manuali. I sistemi di gestione della configurazione forniscono garanzie per quanto riguarda la governance, la conformità e i requisiti normativi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I sistemi di gestione della configurazione vengono utilizzati per tenere traccia e implementare le modifiche nelle configurazioni delle applicazioni e degli ambienti. I sistemi di gestione della configurazione vengono utilizzati anche per ridurre gli errori causati dai processi manuali, rendere le modifiche alla configurazione ripetibili e verificabili e per ridurre il livello di impegno.

AWS ti permette di usare AWS Config per monitorare in modo continuo le configurazioni delle risorse AWS in più account e regioni. Questa soluzione aiuta a tenere traccia della cronologia delle configurazioni, a capire che effetto avrebbe la modifica di una configurazione sulle altre risorse e a verificarle rispetto alle configurazioni previste o desiderate tramite Regole di AWS Config e pacchetti di conformità AWS Config.

Per le configurazioni dinamiche nelle applicazioni in esecuzione su istanze Amazon EC2, AWS Lambda, container, applicazioni mobili o dispositivi IoT, puoi usare AWS AppConfig per configurarle, convalidarle, implementarle e monitorarle nei tuoi ambienti.

Passaggi dell'implementazione

1. Identifica i proprietari della configurazione.
    a. Metti a conoscenza i proprietari delle configurazioni di qualsiasi esigenza di conformità, governance o normativa.
2. Identifica gli elementi e i risultati della configurazione.
    a. Gli elementi di configurazione sono tutte le configurazioni ambientali e dell'applicazione interessate da un'implementazione all'interno della pipeline CI/CD.
    b. I risultati finali includono criteri di successo, convalide e aspetti da monitorare.
3. Seleziona gli strumenti per la gestione della configurazione in base ai requisiti aziendali e alla pipeline di distribuzione.
4. Per modifiche significative alla configurazione, prendi in considerazione le implementazioni ponderate, ad esempio le distribuzioni canary, per ridurre al minimo l'impatto di configurazioni errate.

5. Integra la gestione della configurazione nella tua pipeline CI/CD.
6. Convalida tutte le modifiche inserite.

Risorse

Best practice correlate:

- OPS06-BP01 Piano per modifiche non riuscite
- OPS06-BP02 Implementazioni di test
- OPS06-BP03 Utilizza strategie di implementazione sicure
- OPS06-BP04 Automatizza i test e il rollback

Documenti correlati:

- AWS Control Tower
- AWS Landing Zone Accelerator
- AWS Config
- What is AWS Config?
- AWS AppConfig
- What is AWS CloudFormation?
- Strumenti per sviluppatori in AWS
- AWS CodeBuild
- AWS CodePipeline
- AWS CodeDeploy

Video correlati:

- AWS re:Invent 2022 - Proactive governance and compliance for AWS workloads
- AWS re:Invent 2020: Achieve compliance as code using AWS Config
- Manage and Deploy Application Configurations with AWS AppConfig

OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione

Utilizza sistemi di gestione della creazione e implementazione. Questi sistemi riducono gli errori causati dai processi manuali e il livello di impegno richiesto per la distribuzione delle modifiche.

In AWS, è possibile creare pipeline di integrazione continua/distribuzione continua (CI/CD) con servizi come gli strumenti per sviluppatori AWS (ad esempio, AWS CodeBuild, AWS CodePipeline e AWS CodeDeploy).

Risultato desiderato: i sistemi di gestione della costruzione e dell'implementazione supportano il sistema di distribuzione e integrazione continua (CI/CD) dell'organizzazione, che fornisce funzionalità per automatizzare rollout sicuri con le configurazioni corrette.

Anti-pattern comuni:

- Dopo aver compilato il codice nel sistema di sviluppo, copi il file eseguibile nei sistemi di produzione e questo non si avvia. I file di log locali indicano che l'operazione è risultata impossibile a causa della mancanza di dipendenze.
- Hai creato l'applicazione con nuove funzionalità nel tuo ambiente di sviluppo e fornisci il codice per eseguire il controllo qualità (QA). Il controllo qualità non riesce perché mancano asset statici.
- Venerdì, dopo un notevole sforzo, hai creato l'applicazione manualmente nel tuo ambiente di sviluppo, incluse le nuove funzionalità codificate. Lunedì non sei in grado di ripetere le fasi che ti hanno consentito di creare correttamente la tua applicazione.
- Esegui i test creati per la nuova versione. Quindi passi la settimana successiva a configurare un ambiente di test ed eseguire tutti i test di integrazione esistenti seguiti dai test delle prestazioni. Il nuovo codice ha un impatto inaccettabile sulle prestazioni e deve essere risviluppato e quindi ritestato.

Vantaggi dell'adozione di questa best practice fornendo meccanismi per gestire le attività di compilazione e implementazione, riduci il livello di impegno necessario per eseguire attività ripetitive, consenti ai membri del team di concentrarsi liberamente sulle loro attività creative di valore elevato e limiti l'introduzione di errori derivanti da procedure manuali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I sistemi di gestione della creazione e implementazione vengono utilizzati per tenere traccia e implementare le modifiche, ridurre gli errori causati dai processi manuali e diminuire il livello

di impegno richiesto per le implementazioni sicure. Automatizza completamente la pipeline di integrazione e implementazione dal check-in del codice fino alle fasi di creazione, test, implementazione e convalida. Ciò riduce il lead time e i costi, incoraggia una maggiore frequenza delle modifiche, riduce il livello di impegno e aumenta la collaborazione.

Passaggi dell'implementazione

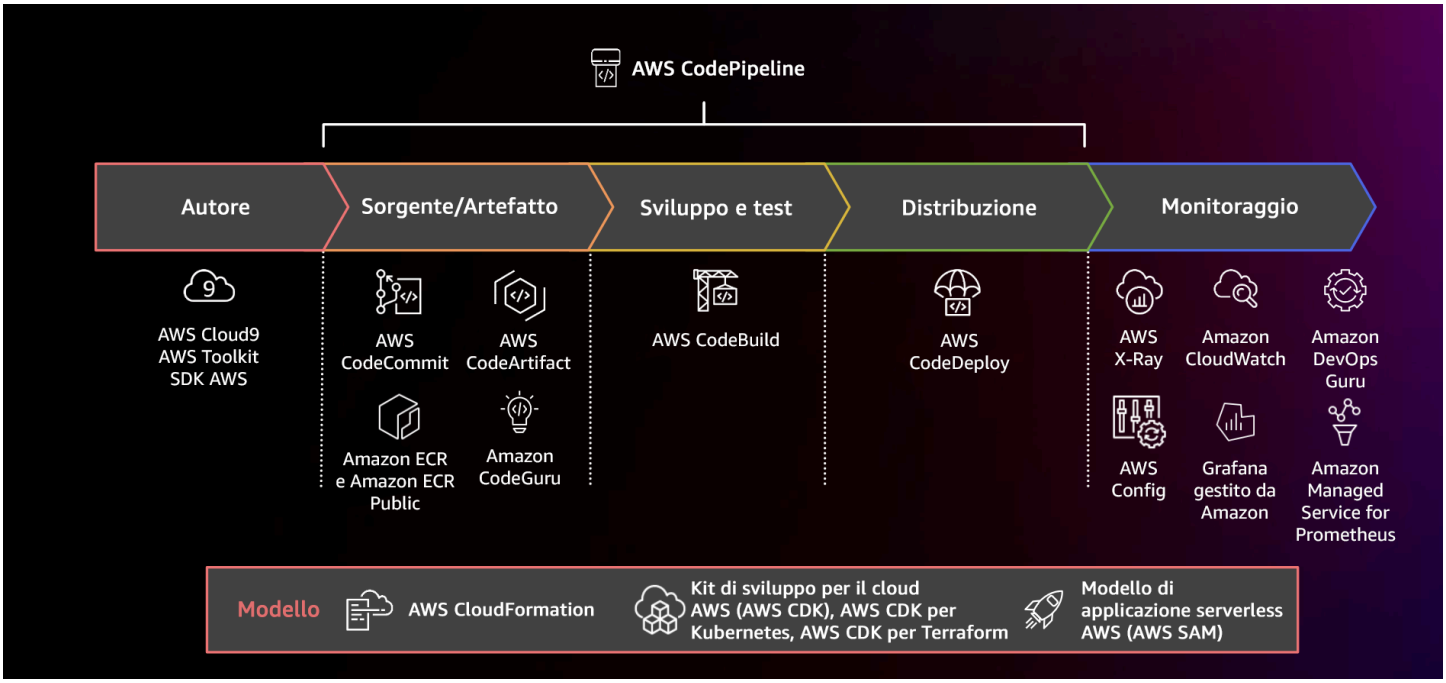


Diagramma che mostra una pipeline CI/CD che utilizza AWS CodePipeline e servizi correlati

1. Utilizza un sistema di controllo delle versioni per archiviare e gestire risorse come documenti, codice sorgente e file binari.
2. Usa CodeBuild per compilare il codice sorgente, esegue unit test e prepara artefatti pronti per essere implementati.
3. Usa CodeDeploy come un servizio di implementazione che automatizza l'implementazione dell'applicazione a istanze Amazon EC2, istanze on-premises, funzioni AWS Lambda serverless o Amazon ECS.
4. Monitora le tue implementazioni.

Risorse

Best practice correlate:

- OPS06-BP04 Automatizza i test e il rollback

Documenti correlati:

- Strumenti per sviluppatori in AWS
- Che cos'è AWS CodeBuild?
- AWS CodeBuild
- Che cos'è AWS CodeDeploy?

Video correlati:

- AWS re:Invent 2022 - AWS Well-Architected best practices for DevOps on AWS

OPS05-BP05 Esecuzione della gestione delle patch

La gestione delle patch consente di ottenere funzionalità, risolvere problemi e rispettare i requisiti di governance. Automatizza la gestione delle patch per ridurre gli errori causati dai processi manuali, dimensionare e ridurre il livello di impegno richiesto per applicare le patch.

La gestione delle patch e delle vulnerabilità fa parte delle attività di gestione dei rischi e dei vantaggi. È preferibile disporre di infrastrutture immutabili e distribuire carichi di lavoro in stati noti verificati. Se ciò non è realizzabile, l'applicazione di patch sul posto è l'alternativa.

AWS Health è la fonte autorevole di informazioni sugli eventi pianificati del ciclo di vita e su altri eventi che richiedono operazioni che influiscono sullo stato delle risorse Cloud AWS. È necessario essere consapevoli delle modifiche e degli aggiornamenti imminenti da eseguire. I principali eventi pianificati relativi al ciclo di vita vengono inviati con almeno sei mesi di anticipo.

Amazon EC2 Image Builder offre pipeline per l'aggiornamento delle immagini delle macchine. Nell'ambito della gestione delle patch, prendi in considerazione Amazon Machine Image (AMI) che utilizza una pipeline di immagini AMI o immagini di container con una pipeline di immagini Docker, mentre AWS Lambda fornisce modelli per runtime personalizzati e librerie aggiuntive in modo da rimuovere le vulnerabilità.

Dovresti gestire gli aggiornamenti di Amazon Machine Image per le immagini di Linux o Windows server mediante Amazon EC2 Image Builder. Puoi utilizzare Amazon Elastic Container Registry (Amazon ECR) con la pipeline esistente per la gestione delle immagini Amazon ECS e Amazon EKS. Lamba offre funzionalità di gestione delle versioni.

L'applicazione di patch non deve essere eseguita sui sistemi di produzione senza prima eseguire test in un ambiente sicuro. Le patch devono essere applicate solo se supportano risultati operativi

o aziendali. AWS offre AWS Systems Manager Patch Manager per automatizzare il processo di applicazione delle patch ai sistemi gestiti e pianificare l'attività mediante le finestre di manutenzione di Systems Manager.

Risultato desiderato: le immagini AMI e dei container sono aggiornate, dotate di patch e pronte per il lancio. È possibile tenere traccia dello stato di tutte le immagini implementate e conoscere la conformità delle patch. Puoi eseguire report sullo stato attuale e disporre di un processo per soddisfare le tue esigenze di conformità.

Anti-pattern comuni:

- Ti viene assegnato il compito di applicare tutte le nuove patch di sicurezza entro 2 ore, il che provoca più interruzioni a causa dell'incompatibilità dell'applicazione con le patch.
- Una libreria senza patch comporta conseguenze indesiderate in quanto parti sconosciute utilizzano vulnerabilità al suo interno per accedere al carico di lavoro.
- L'applicazione di patch agli ambienti per sviluppatori viene eseguita automaticamente senza avvisare gli sviluppatori. Gli sviluppatori ti inviano più reclami perché il loro ambiente non funziona come previsto.
- Non hai applicato patch al software pronto all'uso commerciale su un'istanza persistente. Quando hai problemi con il software e contatti il fornitore, questo ti informa che la versione non è supportata e che devi applicare le patch a un livello specifico per ricevere assistenza.
- Una patch rilasciata di recente per il software di crittografia utilizzato offre miglioramenti significativi in termini di prestazioni. Il sistema privo di patch presenta problemi di prestazioni che rimangono in vigore a causa della mancata applicazione di patch.
- Ricevi una notifica di una vulnerabilità zero-day che richiede una correzione di emergenza; quindi devi applicare manualmente le patch a tutti i tuoi ambienti.
- Non sei a conoscenza delle operazioni critiche necessarie per gestire le risorse, come gli aggiornamenti obbligatori delle versioni, perché non esamini gli eventi pianificati imminenti del ciclo di vita e altre informazioni. Perdi tempo prezioso per la pianificazione e l'esecuzione, con conseguenti modifiche di emergenza per i team e potenziali impatti o tempi di inattività imprevisti.

Vantaggi dell'adozione di questa best practice: stabilendo un processo di gestione delle patch, inclusi i criteri per l'applicazione di patch e la metodologia di distribuzione tra gli ambienti, sarai in grado di dimensionare e generare report sui livelli di patch. Ciò fornisce garanzie sull'applicazione delle patch di sicurezza e una chiara visibilità sullo stato delle correzioni note in atto. Ciò incoraggia l'adozione delle caratteristiche e funzionalità desiderate, aiuta a eliminare rapidamente i problemi e a mantenere

la conformità alla governance. Implementa sistemi di gestione delle patch e automazione per ridurre il livello di impegno per distribuire le patch e limitare gli errori causati dai processi manuali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Applica patch ai sistemi per correggere gli errori, ottenere le funzionalità o le capacità desiderate e assicurare la conformità alle policy di governance e ai requisiti di supporto del fornitore. Nei sistemi immutabili, distribuisci con il set di patch appropriato per raggiungere il risultato desiderato. Automatizza il meccanismo di gestione delle patch per ridurre il tempo necessario per applicare le patch, evitare gli errori causati dai processi manuali e diminuire il livello di impegno richiesto per applicare le patch.

Passaggi dell'implementazione

Per Amazon EC2 Image Builder:

1. Specifica i dettagli della pipeline utilizzando Amazon EC2 Image Builder:
    a. Crea una pipeline di immagini e assegnale un nome.
    b. Definisci la pianificazione e il fuso orario della pipeline.
    c. Configura eventuali dipendenze.
2. Scegli una ricetta:
    a. Seleziona una ricetta esistente o creane una nuova.
    b. Seleziona il tipo di immagine.
    c. Assegna un nome e una versione alla tua ricetta.
    d. Seleziona l'immagine di base.
    e. Aggiungi componenti di compilazione e inseriscili nel registro di destinazione.
3. Facoltativo: definisci la configurazione dell'infrastruttura.
4. Facoltativo: definisci le impostazioni di configurazione.
5. Verifica le impostazioni.
6. Mantieni il livello di igiene delle ricette a livelli ottimali.

Per Gestione patch di Systems Manager:

1. Crea una baseline delle patch.

2. Si seleziona un metodo per le operazioni di applicazione di patch.
3. Abilita il report e la scansione della conformità.

Risorse

Best practice correlate:

- OPS06-BP04 Automatizza i test e il rollback

Documenti correlati:

- What is Amazon EC2 Image Builder
- Create an image pipeline using the Amazon EC2 Image Builder
- Create a container image pipeline
- AWS Systems Manager Patch Manager
- Working with Patch Manager
- Working with patch compliance reports
- Strumenti per sviluppatori in AWS

Video correlati:

- CI/CD for Serverless Applications on AWS
- Design with Ops in Mind

Esempi correlati:

- AWS Systems Manager Patch Manager tutorials

OPS05-BP06 Condividi gli standard di progettazione

Condividi le best practice con i team per incrementare la consapevolezza e potenziare al massimo i vantaggi delle attività di sviluppo. Documentale e mantienile aggiornate di pari passo con l'evoluzione dell'architettura. Se nella tua organizzazione vengono applicati standard condivisi, è fondamentale che esistano meccanismi per richiedere aggiunte, modifiche ed eccezioni agli standard. Senza questa opzione, gli standard diventano un ostacolo per l'innovazione.

Risultato desiderato: gli standard di progettazione vengono condivisi fra team nelle organizzazioni. Sono documentati e conservati up-to-date man mano che le migliori pratiche si evolvono.

Anti-pattern comuni:

- Due team di sviluppo hanno creato ciascuno un servizio di autenticazione utente. Gli utenti devono mantenere un set separato di credenziali per ogni parte del sistema a cui vogliono accedere.
- Ogni team gestisce la propria infrastruttura. Un nuovo requisito di conformità impone una modifica all'infrastruttura e ogni team la implementa in modo diverso.

Vantaggi dell'adozione di questa best practice: l'uso di standard condivisi incoraggia l'applicazione di best practice e permette di ottenere i massimi vantaggi dalle attività di sviluppo. La documentazione e l'aggiornamento degli standard di progettazione consentono all'organizzazione di attenersi up-to-date alle migliori pratiche e ai requisiti di sicurezza e conformità.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Condividi le best practice, gli standard di progettazione, gli elenchi di controllo, le procedure operative, le linee guida e i requisiti di governance esistenti tra team diversi. Definisci procedure per richiedere modifiche, aggiunte ed eccezioni agli standard di progettazione per supportare il miglioramento e l'innovazione. Rendi noto ai team il contenuto pubblicato. Disponete di un meccanismo per mantenere gli standard di progettazione up-to-date man mano che emergono nuove best practice.

Esempio del cliente

AnyCompany Retail dispone di un team di architettura interfunzionale che crea modelli di architettura software. Questo team crea l'architettura con conformità e governance integrate. I team che adottano gli standard condivisi traggono vantaggio dall'integrazione di conformità e governance. Possono creare rapidamente soluzioni sulla base degli standard di progettazione. Il team responsabile dell'architettura si incontra ogni trimestre per valutare i modelli architetturali e aggiornarli, se necessario.

Passaggi dell'implementazione

1. Identifica un team interfunzionale responsabile dello sviluppo e dell'aggiornamento degli standard di progettazione. Questo team collaborerà con le parti interessate in tutta l'organizzazione per

sviluppare standard di progettazione, procedure operative, elenchi di controllo, linee guida e requisiti di governance. Documenta gli standard di progettazione e condividili internamente all'organizzazione.

a. AWS Service Catalog può aiutarti a creare portfolio che rappresentano gli standard di progettazione usando il modello Infrastructure as code (IaC). Puoi condividere portfolio tra più account.

2. Disponete di un meccanismo per mantenere gli standard di progettazione up-to-date man mano che vengono identificate nuove best practice.
3. Se gli standard di progettazione vengono applicati a livello centrale, definisci un processo per richiedere modifiche, aggiornamenti ed eccezioni.

Livello di impegno per il piano di implementazione: medio Lo sviluppo di un processo per creare e condividere standard di progettazione può richiedere il coordinamento e la cooperazione con le parti interessate in tutta l'organizzazione.

Risorse

Best practice correlate:

- OPS01-BP03 Valuta i requisiti di governance: i requisiti di governance influiscono sugli standard di progettazione.
- OPS01-BP04 Valutazione dei requisiti di conformità: la conformità è un fattore essenziale nella creazione di standard di progettazione.
- OPS07-BP02 Revisione costante della prontezza operativa: gli elenchi di controllo della prontezza operativa sono un meccanismo per implementare standard di progettazione durante la progettazione del carico di lavoro.
- OPS11-BP01 Definizione di un processo per il miglioramento continuo: l'aggiornamento degli standard di progettazione contribuisce a un miglioramento continuo.
- OPS11-BP04 Eseguire la gestione della conoscenza: nell'ambito della procedura di gestione delle informazioni, documenta e condividi gli standard di progettazione.

Documenti correlati:

- AWS Backup Automatizzaci con AWS Service Catalog
- AWS Service Catalog Account Factory-Enhanced

- In che modo Expedia Group ha creato l'offerta Database as a Service () utilizzando DBaaS AWS Service Catalog
- Maintain visibility over the use of cloud architecture patterns
- Semplifica la condivisione dei tuoi AWS Service Catalog portafogli in un'unica configurazione AWS Organizations

Video correlati:

- AWS Service Catalog — Guida introduttiva
- AWS re:Invent 2020: gestisci i tuoi AWS Service Catalog portafogli come un esperto

Esempi correlati:

- AWS Service Catalog Architettura di riferimento
- AWS Service Catalog Workshop

Servizi correlati:

- AWS Service Catalog

OPS05-BP07 Implementazione di prassi per migliorare la qualità del codice

Implementa prassi per migliorare la qualità del codice e ridurre al minimo i difetti. Alcuni esempi includono sviluppo basato su test, revisioni del codice, adozione degli standard e programmazione in coppia. Inserisci queste prassi nel processo di integrazione continua e distribuzione continua.

Risultato desiderato: la tua organizzazione usa best practice come le revisioni del codice e la programmazione in coppia per migliorare la qualità del codice. Sviluppatori e operatori adottano le best practice per la qualità del codice nell'ambito del ciclo di vita di sviluppo del software.

Anti-pattern comuni:

- Commit del codice nel ramo principale dell'applicazione senza alcuna revisione. In questo modo, la modifica viene implementata in automatico nell'ambiente di produzione e causa un'interruzione.
- Sviluppo di una nuova applicazione senza unit test, test end-to-end o test di integrazione. Non è possibile in alcun modo testare l'applicazione prima dell'implementazione.

- I team apportano modifiche manuali nell'ambiente di produzione per gestire gli errori. Le modifiche non vengono sottoposte a test o revisioni del codice, né vengono acquisite o registrate durante i processi di integrazione continua e distribuzione continua.

Vantaggi dell'adozione di questa best practice: l'adozione di pratiche per migliorare la qualità del codice ti consente di ridurre al minimo i problemi di produzione. Le best practice per la qualità del codice includono la programmazione in coppia, le revisioni del codice e l'implementazione di strumenti di produttività basati sull'IA.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Implementa prassi per migliorare la qualità del codice in modo da ridurre gli errori prima dell'implementazione. Usa prassi come lo sviluppo basato su test, le revisioni del codice e la programmazione in coppia per migliorare la qualità dello sviluppo.

Usa la potenza dell'IA generativa con Amazon Q Developer per migliorare la produttività degli sviluppatori e la qualità del codice. Amazon Q Developer include la generazione di suggerimenti di codice (basati su modelli linguistici di grandi dimensioni), la produzione di test di unità (comprese le condizioni limite) e il miglioramento della sicurezza del codice tramite il rilevamento e la correzione delle vulnerabilità di sicurezza.

Esempio del cliente

AnyCompany Retail adotta diverse prassi per migliorare la qualità del codice. L'azienda ha adottato lo sviluppo basato su test come standard per la scrittura di applicazioni. Per alcune nuove funzionalità, gli sviluppatori eseguiranno la programmazione in coppia durante uno sprint. Ogni richiesta pull viene sottoposta a una revisione del codice da parte di uno sviluppatore senior prima di essere integrata e implementata.

Passaggi dell'implementazione

1. Adotta prassi per la qualità del codice come lo sviluppo basato su test, le revisioni del codice e la programmazione in coppia nel processo di integrazione continua e distribuzione continua. Usa queste tecniche per migliorare la qualità del software.
   a. Usa Amazon Q Developer, uno strumento di IA generativa utile per creare casi di test di unità (comprese le condizioni limite), generare funzioni utilizzando codice e commenti, implementare algoritmi noti, rilevare violazioni e vulnerabilità delle policy di sicurezza nel codice, rilevare

segreti, effettuare la scansione dell'infrastructure as code (IaC), documentare il codice e apprendere più rapidamente le librerie di codici di terze parti.

b. Il revisore Amazon CodeGuru può fornire suggerimenti di programmazione per il codice Java e Python tramite il machine learning.

Livello di impegno per il piano di implementazione: medio Esistono molti modi per implementare questa best practice, ma la realizzazione dell'adozione da parte dell'organizzazione può essere problematica.

Risorse

Best practice correlate:

- OPS05-BP02 Test e convalida delle modifiche
- OPS05-BP06 Condivisione degli standard di progettazione

Documenti correlati:

- Adozione di un approccio di sviluppo basato su test
- Accelerate your Software Development Lifecycle with Amazon Q
- Amazon Q Developer, now generally available, includes previews of new capabilities to reimagine developer experience
- The Ultimate Cheat Sheet for Using Amazon Q Developer in Your IDE
- Shift-Left Workload, leveraging AI for Test Creation
- Amazon Q Developer Center
- 10 ways to build applications faster with Amazon CodeWhisperer
- Looking beyond code coverage with Amazon CodeWhisperer
- Best Practices for Prompt Engineering with Amazon CodeWhisperer
- Agile Software Guide
- My CI/CD pipeline is my release captain
- Automate code reviews with Amazon CodeGuru Reviewer
- Adozione di un approccio di sviluppo basato su test
- How DevFactory builds better applications with Amazon CodeGuru
- On Pair Programming

- RENGA Inc. automates code reviews with Amazon CodeGuru
- The Art of Agile Development: Test-Driven Development
- Why code reviews matter (and actually save time!)

Video correlati:

- Implement an API with Amazon Q Developer Agent for Software Development
- Installing, Configuring, & Using Amazon Q Developer with JetBrains IDEs (How-to)
- Mastering the art of Amazon CodeWhisperer - YouTube playlist
- AWS re:Invent 2020: Continuous improvement of code quality with Amazon CodeGuru
- AWS Summit ANZ 2021 - Driving a test-first strategy with CDK and test driven development

Servizi correlati:

- Amazon Q Developer
- Revisore Amazon CodeGuru
- Profilatore Amazon CodeGuru

OPS05-BP08 Utilizzo di più ambienti

Utilizza più ambienti per sperimentare, sviluppare e testare il carico di lavoro. Applica livelli crescenti di controlli man mano che gli ambienti si avvicinano alla fase di produzione per avere la certezza che il carico di lavoro funzioni come previsto una volta implementato.

Risultato desiderato: disponi di più ambienti che riflettono le tue esigenze di conformità e governance. Testi e promuovi il codice negli ambienti lungo il tuo percorso verso la produzione.

1. L'organizzazione esegue queste operazioni attraverso la creazione di una zona di destinazione, che fornisce governance, controlli, automazioni degli account, rete, sicurezza e osservabilità operativa. Gestisci queste funzionalità di zona di destinazione utilizzando più ambienti. Un esempio comune è un'organizzazione sandbox per lo sviluppo e il test delle modifiche apportate a una zona di destinazione basata su AWS Control Tower, che include AWS IAM Identity Center e policy quali le policy di controllo dei servizi. Tutti questi elementi possono avere un impatto significativo sull'accesso e sul funzionamento degli Account AWS all'interno della zona di destinazione.

2. Oltre a questi servizi, i team possono estendere le capacità delle zone di destinazione con soluzioni pubblicate da AWS e dai partner AWS o come soluzioni personalizzate sviluppate all'interno dell'organizzazione. Esempi di soluzioni pubblicate da AWS includono Customizations for AWS Control Tower (CfCT) e AWS Control Tower Account Factory for Terraform (AFT).
3. L'organizzazione applica gli stessi principi di test, promozione del codice e modifiche alle policy per la zona di destinazione attraverso gli ambienti nel percorso verso la produzione. Questa strategia fornisce un ambiente di zona di destinazione stabile e sicuro per i team delle applicazioni e dei carichi di lavoro.

Anti-pattern comuni:

- Stai sviluppando in un ambiente di sviluppo condiviso e un altro sviluppatore sovrascrive le tue modifiche al codice.
- I controlli di sicurezza restrittivi nell'ambiente di sviluppo condiviso impediscono di sperimentare nuovi servizi e funzionalità.
- Esegui test di carico sui tuoi sistemi di produzione e causa un'interruzione per i tuoi utenti.
- Si è verificato un errore critico che ha causato la perdita di dati nella produzione. Nel tuo ambiente di produzione tenti di ricreare le condizioni che portano alla perdita di dati in modo da poter identificare come si è verificata e impedire che si ripeta. Per evitare un'ulteriore perdita di dati durante il test, devi rendere l'applicazione non disponibile per i tuoi utenti.
- Stai operando un servizio multi-tenant e non sei in grado di supportare la richiesta di un cliente per un ambiente dedicato.
- Ogni volta che esegui un test, lo fai nel tuo ambiente di produzione.
- Ritieni che la semplicità di un singolo ambiente prevalga sulla portata dell'impatto che possono avere modifiche all'interno dell'ambiente.
- Aggiorni una funzionalità chiave della zona di destinazione ma la modifica compromette la capacità del team di vendere gli account per nuovi progetti o per i carichi di lavoro esistenti.
- Applichi nuovi controlli agli Account AWS, ma la modifica ha un impatto sulla capacità del team del carico di lavoro di implementare le modifiche all'interno dei propri Account AWS.

Vantaggi dell'adozione di questa best practice: quando distribuisci più ambienti, puoi supportare più ambienti di sviluppo, test e produzione simultanei senza creare conflitti tra sviluppatori o community di utenti. Per funzionalità complesse come le zone di destinazione, riduce in modo significativo il rischio di modifiche, semplifica il processo di miglioramento e riduce il rischio di aggiornamenti critici

dell'ambiente. Le organizzazioni che utilizzano le zone di destinazione beneficiano naturalmente di più account nel loro ambiente AWS, con struttura dell'account, governance, rete e configurazioni di sicurezza. Nel corso del tempo, con la crescita dell'organizzazione, la zona di destinazione può evolvere per proteggere e organizzare i carichi di lavoro e le risorse.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Utilizza più ambienti e fornisci agli sviluppatori ambienti sandbox con controlli minimi per incoraggiare la sperimentazione. Fornisci ambienti di sviluppo individuali per facilitare il lavoro in parallelo, incrementando l'agilità dello sviluppo. Implementa controlli più rigorosi negli ambienti che si avvicinano alla produzione per consentire agli sviluppatori di innovare. Utilizza l'approccio Infrastructure as code e sistemi di gestione delle configurazioni per distribuire ambienti configurati in modo coerente con i controlli presenti in produzione per assicurare che i sistemi funzionino nel modo previsto quando vengono distribuiti. Quando gli ambienti non vengono utilizzati, disattivali per evitare costi associati alle risorse inattive, ad esempio i sistemi di sviluppo nelle ore serali e nei fine settimana. Durante i test di carico, è necessario implementare ambienti equivalenti a quelli di produzione per migliorare la validità dei risultati.

Team come l'ingegneria della piattaforma, la rete e le operazioni di sicurezza spesso gestiscono le funzionalità a livello di organizzazione con requisiti distinti. La sola separazione degli account non è sufficiente a fornire e mantenere ambienti separati per la sperimentazione, lo sviluppo e i test. In questi casi, crea istanze separate di AWS Organizations.

Risorse

Documenti correlati:

- Pianificatore di istanze su AWS
- Che cos'è AWS CloudFormation?
- Organizing Your AWS Environment Using Multiple Accounts - Multiple organizations - Test changes to your overall AWS environment
- AWS Control Tower Guide

OPS05-BP09 Apporta modifiche frequenti, piccole e reversibili

Le modifiche frequenti, minime e reversibili riducono la portata e l'impatto di una modifica. Le modifiche frequenti, minime e reversibili, se effettuate utilizzando congiuntamente sistemi di gestione

delle modifiche, di gestione della configurazione e di compilazione e distribuzione, riducono la portata e l'impatto di una modifica. Questo si traduce in una risoluzione dei problemi più efficace, accelerando la correzione e mantenendo la possibilità di rollback delle modifiche.

Anti-pattern comuni:

- Distribuisci una nuova versione della tua applicazione ogni trimestre con una finestra di modifica, il che comporta la disattivazione di un servizio di base.
- Spesso apporti modifiche allo schema del database senza che ne venga tenuta traccia nei sistemi di gestione.
- Esegui aggiornamenti manuali sul posto, sovrascrivendo le installazioni e le configurazioni esistenti, senza avere un chiaro piano di rollback.

Vantaggi dell'adozione di questa best practice: velocizzazione degli sforzi di sviluppo grazie all'implementazione frequente di piccole modifiche. Quando le modifiche sono minime, è molto più semplice identificare se hanno conseguenze indesiderate e, in tal caso, ripristinare la condizione precedente. Quando le modifiche sono reversibili, il rischio di implementare le modifiche è minore in quanto il ripristino è semplificato. Il processo di modifica comporta un rischio ridotto e l'impatto di una modifica non corretta è ridotto.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Applica modifiche frequenti, minime e reversibili per ridurre la portata e l'impatto di una modifica. In questo modo si semplifica la risoluzione dei problemi, si velocizza la correzione ed è possibile eseguire il rollback di una modifica. Inoltre, aggiunge più rapidamente valore al business.

Risorse

Best practice correlate:

- OPS05-BP03 Utilizzo di sistemi di gestione delle configurazioni
- OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione
- OPS06-BP04 Automatizza i test e il rollback

Documenti correlati:

- Implementazione di microservizi su AWS
- Microservices - Observability

OPS05-BP10 Automazione completa dell'integrazione e dell'implementazione

Automatizza la creazione, l'implementazione e il test del carico di lavoro. Questo riduce gli errori causati dai processi manuali e l'impegno necessario per distribuire le modifiche.

Applica i metadati utilizzando i tag delle risorse e gli AWS Resource Groups seguendo una strategia di applicazione dei tag coerente per consentire l'identificazione delle risorse. Applica tag alle risorse per organizzare, monitorare i costi e controllare gli accessi e ottimizza l'esecuzione delle attività operative automatizzate.

Risultato desiderato: chi si occupa di sviluppo utilizza strumenti per distribuire codice ed effettuare la promozione a produzione. Gli sviluppatori non devono effettuare il login alla Console di gestione AWS per fornire gli aggiornamenti. Esiste un audit trail completo di modifiche e configurazioni che soddisfa le esigenze di governance e conformità. I processi sono ripetibili e standardizzati tra i team. Gli sviluppatori sono liberi di concentrarsi sullo sviluppo e sui rilasci del codice, aumentando la produttività.

Anti-pattern comuni:

- Venerdì termini la creazione del nuovo codice per il ramo delle funzionalità. Lunedì, dopo aver eseguito gli script di test di qualità del codice e tutti gli script dei test di unità, effettui il check-in del codice per il prossimo rilascio programmato.
- Ti verrà assegnato di codificare una correzione per un problema critico che interessa un numero elevato di clienti nella produzione. Dopo aver testato la correzione, esegui il commit del codice e richiedi via e-mail alla gestione delle modifiche l'approvazione per implementarlo in produzione.
- In qualità di sviluppatore, accedi alla Console di gestione AWS per creare un nuovo ambiente di sviluppo utilizzando metodi e sistemi non standard.

Vantaggi dell'adozione di questa best practice: implementando sistemi di gestione automatizzati di compilazione e implementazione, si riduce il numero di errori causati dai processi manuali e lo sforzo di implementare le modifiche aiutando i membri del team a concentrarsi sull'offerta di valore aggiunto. Maggiore velocità di consegna man mano che procedi verso la promozione a produzione.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Utilizza i sistemi di gestione della compilazione e implementazione per tenere traccia e implementare le modifiche, ridurre gli errori causati dai processi manuali e ridurre il livello di impegno richiesto. Automatizza completamente la pipeline di integrazione e implementazione dal check-in del codice fino alle fasi di creazione, test, implementazione e convalida. In questo modo è possibile diminuire il lead time, incoraggiare una maggiore frequenza di modifica, ridurre il livello di impegno e accelerare il time-to-market, il che si traduce in una maggiore produttività e in un aumento della sicurezza del codice man mano che procedi con la promozione verso la produzione.

Risorse

Best practice correlate:

- OPS05-BP03 Utilizzo di sistemi di gestione delle configurazioni
- OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione

Documenti correlati:

- Che cos'è AWS CodeBuild?
- Che cos'è AWS CodeDeploy?

Video correlati:

- AWS re:Invent 2022 - AWS Well-Architected best practices for DevOps on AWS

OPS 6. In che modo mitighi i rischi dell'implementazione?

Adotta approcci per fornire un feedback rapido sulla qualità e che permettano un ripristino veloce dalle modifiche che non hanno i risultati previsti. L'uso di queste prassi consente di mitigare l'impatto dei problemi introdotti attraverso l'implementazione delle modifiche.

Best practice

- OPS06-BP01 Piano per modifiche non riuscite
- OPS06-BP02 Implementazioni di test
- OPS06-BP03 Utilizza strategie di implementazione sicure
- OPS06-BP04 Automatizza i test e il rollback

OPS06-BP01 Piano per modifiche non riuscite

Pianifica il ripristino di uno stato corretto noto o la correzione nell'ambiente di produzione nel caso in cui l'implementazione generi un risultato indesiderato. Disporre di una policy per stabilire un piano di questo tipo aiuta tutti i team a sviluppare strategie di ripristino dalle modifiche con esito negativo. Alcune strategie di esempio sono le fasi di implementazione e rollback, le policy di modifica, i flag di funzionalità, l'isolamento del traffico e lo spostamento del traffico. Una singola release può includere più modifiche ai componenti correlati. La strategia dovrebbe fornire la capacità di resistere o ripristinare in caso di guasto generato da qualsiasi modifica dei componenti.

Risultato desiderato: hai preparato un piano di ripristino dettagliato per la modifica in caso di fallimento. Inoltre, hai ridotto le dimensioni della release per ridurre al minimo il potenziale impatto su altri componenti del carico di lavoro. Di conseguenza, hai ridotto l'impatto aziendale abbreviando i potenziali tempi di inattività causati da una modifica non riuscita e aumentando la flessibilità e l'efficienza dei tempi di ripristino.

Anti-pattern comuni:

- Hai eseguito un'implementazione e l'applicazione è diventata instabile, ma sembra che ci siano utenti attivi sul sistema. Devi decidere se eseguire il rollback della modifica e influire sugli utenti attivi o aspettare di eseguire il rollback della modifica, sapendo che gli utenti potranno essere comunque influenzati.
- Dopo aver apportato una modifica di routine, i nuovi ambienti sono accessibili, ma una delle sottoreti è diventata irraggiungibile. Devi decidere se eseguire il rollback di tutto o provare a correggere il problema della sottorete inaccessibile. Mentre prendi tale decisione, la sottorete rimane irraggiungibile.
- I tuoi sistemi non sono progettati in modo da consentire loro di essere aggiornati con release più piccole. Di conseguenza, è difficile annullare tali modifiche di massa (bulk changes) durante un'implementazione conclusasi con esito negativo.
- Non utilizzi il modello Infrastructure as code (IaC) e hai apportato aggiornamenti manuali all'infrastruttura che hanno portato a configurazioni indesiderate. Non è possibile tracciare e ripristinare in modo efficace le modifiche manuali.
- Poiché non hai misurato l'aumento della frequenza delle implementazioni, il tuo team non è incentivato a ridurre le dimensioni delle modifiche e a migliorare i piani di rollback per ogni modifica, con conseguente aumento dei rischi e dei tassi di fallimento.

- Non misuri la durata totale di un'interruzione causata da modifiche con esito negativo. Il tuo team non è in grado di stabilire le priorità e migliorare il processo di implementazione e l'efficacia del piano di ripristino.

Vantaggi derivanti dall'adozione di questa procedura ottimale: disporre di un piano di ripristino in caso di modifiche non riuscite riduce al minimo il tempo medio di ripristino () MTTR e riduce l'impatto aziendale.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Una policy e una pratica coerenti e documentate adottate dai team di rilascio consentono a un'organizzazione di pianificare cosa dovrebbe succedere in caso di modifiche con esito negativo. In circostanze specifiche la policy dovrebbe consentire la possibilità di apportare correzioni per garantire la prosecuzione del processo. In entrambe le situazioni, un piano di correzione (fix forward) o ripristino (rollback) deve essere ben documentato e testato prima dell'implementazione nei sistemi di produzione live, in modo da ridurre al minimo il tempo necessario per ripristinare una modifica.

Passaggi dell'implementazione

1. Documenta le policy che richiedono ai team di disporre di piani efficaci per invertire le modifiche entro un periodo di tempo specificato.
   a. Le policy devono specificare quando è consentita una situazione di applicazione di correzioni per garantire la prosecuzione del processo.
   b. Richiedi un piano di rollback documentato che sia accessibile a tutti i soggetti coinvolti.
   c. Specifica i requisiti per il rollback (ad esempio, quando si rileva che sono state implementate modifiche non autorizzate).
2. Analizza il livello di impatto di tutte le modifiche relative a ciascun componente di un carico di lavoro.
   a. Consenti che le modifiche ripetibili siano standardizzate, basate su modelli e preautorizzate se seguono un flusso di lavoro coerente che applica le policy di modifica.
   b. Riduci il potenziale impatto di qualsiasi modifica riducendone le dimensioni, in modo che il ripristino richieda meno tempo e abbia un impatto aziendale minore.
   c. Assicurati che le procedure di rollback riportino il codice allo stato corretto noto per evitare incidenti, ove possibile.
3. Integra strumenti e flussi di lavoro per applicare le tue policy in modo programmatico.

4. Rendi visibili i dati sulle modifiche agli altri responsabili di carichi di lavoro per migliorare la velocità di diagnosi di eventuali modifiche con esito negativo che non possono essere ripristinate.
    a. Misura il successo di questa pratica utilizzando dati di modifica visibili e identifica miglioramenti iterativi.
5. Utilizza gli strumenti di monitoraggio per verificare il successo o il fallimento di un'implementazione per accelerare il processo decisionale sul rollback.
6. Misura la durata dell'interruzione durante una modifica con esito negativo per migliorare continuamente i tuoi piani di ripristino.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS06-BP04 Automatizza i test e il rollback

Documenti correlati:

- AWS Builders Library | Garantire la sicurezza del rollback durante le implementazioni
- AWS Whitepaper | Gestione delle modifiche nel cloud

Video correlati:

- re:Invent 2019 | Amazon's approach to high-availability deployment

OPS06-BP02 Implementazioni di test

Testa le procedure di rilascio in pre-produzione utilizzando la stessa configurazione di implementazione, i controlli di sicurezza, i passaggi e le procedure utilizzati nell'ambiente di produzione. Verifica che tutte le fasi implementate siano state completate come previsto, ad esempio l'ispezione di file, configurazioni e servizi. Verifica ulteriormente tutte le modifiche con test funzionali, di integrazione e di carico, oltre ad attivare tutte le attività di monitoraggio come i controlli dell'integrità. Eseguendo questi test, è possibile identificare tempestivamente i problemi di implementazione con l'opportunità di pianificarli e mitigarli prima del passaggio nell'ambiente di produzione.

Puoi creare ambienti paralleli temporanei per testare ogni modifica. Automatizza l'implementazione degli ambienti di test utilizzando il modello Infrastructure as code (IaC) per ridurre la quantità di lavoro necessaria e garantire stabilità, coerenza e una distribuzione più rapida delle funzionalità.

Risultato desiderato: la tua organizzazione adotta una cultura di sviluppo che include il test delle implementazioni. Ciò garantisce che i team siano concentrati sulla realizzazione di valore aziendale anziché sulla gestione delle release. I team vengono coinvolti fin dall'identificazione dei rischi di implementazione per determinare il percorso di mitigazione appropriato.

Anti-pattern comuni:

- Durante le release di produzione, le implementazioni non testate causano problemi frequenti che richiedono una risoluzione mirata e l'escalation.
- La tua release contiene porzioni del modello Infrastructure as code (IaC) che aggiornano le risorse esistenti. Non sei sicuro che l'IaC funzionerà correttamente e non avrà un impatto sulle risorse.
- Viene implementata una nuova funzionalità interessante nella tua applicazione. Non funziona come previsto e non c'è visibilità finché non viene segnalata dagli utenti interessati.
- I certificati vengono aggiornati. Si installano accidentalmente i certificati sui componenti sbagliati, il che non viene rilevato e influisce sui visitatori poiché non è possibile stabilire una connessione sicura al sito web.

Vantaggi dell'adozione di questa best practice: test approfonditi in fase di pre-produzione delle procedure di implementazione e delle modifiche da queste introdotte riducono al minimo il potenziale impatto sulla produzione causato dalle fasi di implementazione. Ciò aumenta la fiducia durante il rilascio in produzione e riduce al minimo la necessità di supporto operativo senza rallentare la velocità di distribuzione delle modifiche apportate.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Testare il processo di implementazione è importante quanto testare le modifiche derivanti dall'implementazione. Ciò può essere ottenuto testando le fasi di implementazione in un ambiente di pre-produzione che rispecchi il più fedelmente possibile quello di produzione. I problemi più comuni, come fasi di implementazione incomplete o contenenti errori o configurazioni errate, possono essere individuati di conseguenza prima di passare all'ambiente di produzione. Inoltre, è possibile testare le fasi di ripristino.

Esempio del cliente

Nell'ambito della propria pipeline di integrazione e distribuzione continua (CI/CD), AnyCompany Retail esegue le fasi definite necessarie per rilasciare aggiornamenti dell'infrastruttura e del software per i propri clienti in un ambiente simile a quello di produzione. La pipeline comprende controlli preliminari per rilevare le deviazioni (il rilevamento delle modifiche alle risorse eseguite al di fuori dell'IaC) nelle risorse prima dell'implementazione, nonché per convalidare le azioni che l'IaC intraprende al suo avvio. Convalida le fasi dell'implementazione, ad esempio la verifica che determinati file e configurazioni siano presenti e che i servizi siano in esecuzione e rispondano correttamente ai controlli dell'integrità sull'host locale, prima di effettuare nuovamente la registrazione sul bilanciatore del carico. Inoltre, tutte le modifiche attivano una serie di test automatici, come test funzionali, di sicurezza, di regressione, di integrazione e di carico.

Passaggi dell'implementazione

1. Esegui controlli di pre-installazione per rispecchiare l'ambiente di pre-produzione in produzione.
   a. Utilizza il rilevamento della deriva per rilevare quando le risorse sono state modificate all'esterno. CloudFormation
   b. Utilizzate i set di modifiche per verificare che l'intento di un aggiornamento dello stack corrisponda alle azioni CloudFormation intraprese all'avvio del set di modifiche.
2. Ciò attiva una fase di approvazione manuale in AWS CodePipeline per autorizzare l'implementazione nell'ambiente di preproduzione.
3. Utilizza configurazioni di distribuzione, come AWS CodeDeploy AppSpeci file, per definire le fasi di distribuzione e convalida.
4. Ove applicabile, esegui AWS CodeDeploy l'integrazione con altri AWS servizi o AWS CodeDeploy con prodotti e servizi dei partner.
5. Monitora le distribuzioni utilizzando Amazon e CloudWatch le AWS CloudTrail notifiche SNS degli eventi di Amazon.
6. Esegui test automatici post-implementazione, inclusi test funzionali, di sicurezza, di regressione, di integrazione e di carico.
7. Risoluzione dei problemi relativi alle implementazioni.
8. La corretta convalida dei passaggi precedenti dovrebbe attivare un flusso di lavoro di approvazione manuale per autorizzare l'implementazione nell'ambiente di produzione.

Livello di impegno per il piano di implementazione: elevato

Risorse

Best practice correlate:

- OPS05-BP02 Test e convalida delle modifiche

Documenti correlati:

- AWS Builders' Library | Automatizzazione di implementazioni sicure e pratiche | Distribuzioni di test
- AWS Whitepaper | Praticare l'integrazione continua e la distribuzione continua su AWS
- The Story of Apollo - Amazon's Deployment Engine
- Come eseguire test ed eseguire il debug AWS CodeDeploy localmente prima di spedire il codice
- Integrating Network Connectivity Testing with Infrastructure Deployment

Video correlati:

- re:Invent 2020 | Testing software and systems at Amazon

Esempi correlati:

- Tutorial | Implementazione e ECS servizio Amazon con un test di convalida

OPS06-BP03 Utilizza strategie di implementazione sicure

I roll-out sicuri della produzione controllano il flusso di modifiche vantaggiose con l'obiettivo di ridurre al minimo l'impatto percepito di tali modifiche sui clienti. I controlli di sicurezza forniscono meccanismi di ispezione per convalidare i risultati desiderati e limitare l'ambito di impatto derivante da eventuali difetti introdotti dalle modifiche o da errori di implementazione. I roll-out sicuri possono includere strategie come feature-flags, one-box, roll-out (release canary), immutabili, suddivisioni del traffico e implementazioni blu/verdi.

Risultato desiderato: l'organizzazione utilizza un sistema di distribuzione e integrazione continua (CI/CD) che fornisce funzionalità per automatizzare roll-out sicuri. I team sono tenuti a utilizzare strategie di roll-out sicure appropriate.

Anti-pattern comuni:

- Implementi una modifica non riuscita a tutta la produzione contemporaneamente. Di conseguenza, tutti i clienti vengono colpiti contemporaneamente.
- Un difetto introdotto in un'implementazione simultanea su tutti i sistemi richiede una release di emergenza. La correzione per tutti i clienti richiede diversi giorni.
- La gestione della release di produzione richiede la pianificazione e la partecipazione di diversi team. Ciò limita la tua capacità di aggiornare frequentemente le funzionalità per i tuoi clienti.
- Esegui un'implementazione variabile modificando i sistemi esistenti. Dopo aver scoperto che la modifica non è andata a buon fine, devi modificare nuovamente i sistemi per ripristinare la versione precedente estendendo il tempo di ripristino.

Vantaggi dell'adozione di questa best practice: le implementazioni automatizzate bilanciano la velocità dei roll-out con la fornitura costante di modifiche vantaggiose per i clienti. La limitazione dell'impatto previene costosi errori di implementazione e massimizza la capacità dei team di rispondere in modo efficiente ai guasti.

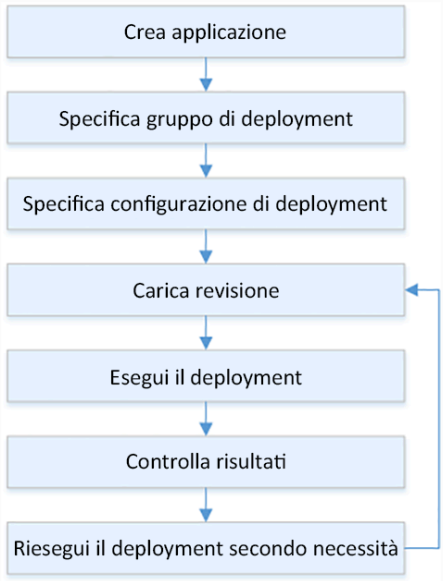
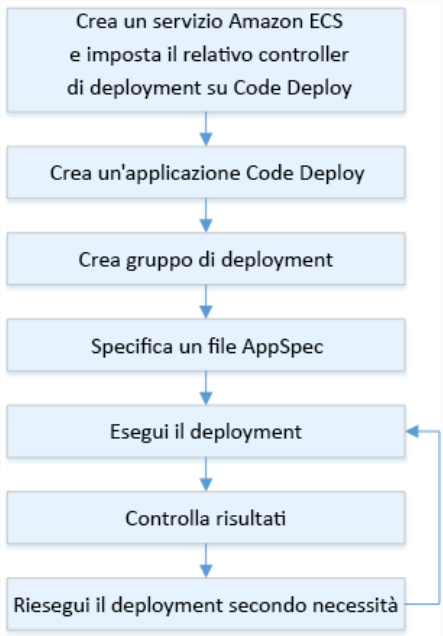
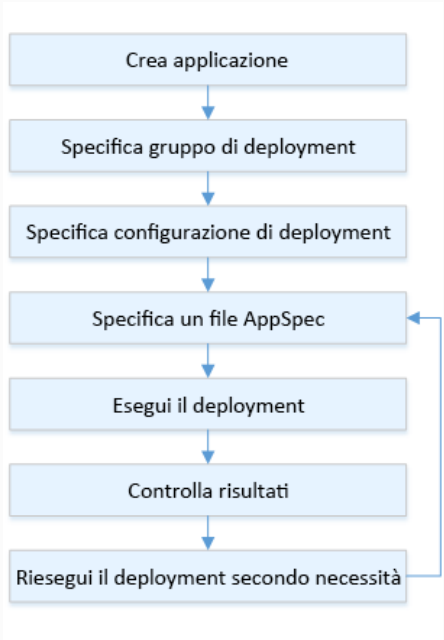
Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Gli errori della distribuzione continua possono portare a una ridotta disponibilità del servizio e a esperienze dei clienti negative. Per massimizzare il tasso di implementazione di successo, implementa i controlli di sicurezza nel processo di rilascio end-to-end per ridurre al minimo gli errori di implementazione, con l'obiettivo di raggiungere il traguardo di zero errori.

Esempio del cliente

La missione di AnyCompany Retail è raggiungere implementazioni con tempi di inattività minimi o pari a zero, il che significa che non vi deve essere alcun impatto percepibile dagli utenti durante le implementazioni. A tal fine, l'azienda ha stabilito modelli di implementazione (vedi il seguente diagramma del flusso di lavoro) come roll-out e implementazioni blu/verdi. Tutti i team adottano uno o più di questi modelli nella loro pipeline CI/CD.

Flusso di lavoro CodeDeploy per Amazon EC2	Flusso di lavoro CodeDeploy per Amazon ECS	Flusso di lavoro CodeDeploy per Lambda
		

Passaggi dell'implementazione

1. Utilizza un flusso di lavoro di approvazione per avviare la sequenza delle fasi di roll-out della produzione al momento della promozione alla produzione.
2. Usa un sistema di implementazione automatizzata come AWS CodeDeploy. Le opzioni di implementazioni AWS CodeDeploy comprendono le implementazioni locali (in-place) per EC2/on-premises e le implementazioni blu/verde per EC2/on-premises. AWS Lambda e Amazon ECS (vedi il diagramma del flusso di lavoro precedente).
   a. Ove applicabile, integra AWS CodeDeploy con altri servizi AWS o con prodotti e servizi dei partner AWS CodeDeploy.
3. Ricorri a implementazioni blu/verde per database come Amazon Aurora e Amazon RDS.
4. Monitora le implementazioni utilizzando Amazon CloudWatch, AWS CloudTrail e le notifiche di eventi di Amazon Simple Notification Service (Amazon SNS).
5. Esegui test automatici post-implementazione, inclusi test funzionali, di sicurezza, di regressione, di integrazione e di carico.
6. Risoluzione dei problemi relativi alle implementazioni.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS05-BP02 Test e convalida delle modifiche
- OPS05-BP09 Apporta modifiche frequenti, piccole e reversibili
- OPS05-BP10 Automazione completa dell'integrazione e dell'implementazione

Documenti correlati:

- AWS Builders Library | Automatizzazione di distribuzioni pratiche e sicure | Distribuzioni di produzione
- AWS Builders Library | My CI/CD pipeline is my release captain | Safe, automatic production releases
- Whitepaper AWS | Integrazione e distribuzione continua in AWS | Metodi di implementazione
- AWS CodeDeploy Guida per l'utente di
- Working with deployment configurations in AWS CodeDeploy
- Set up an API Gateway canary release deployment
- Amazon ECS Deployment Types
- Fully Managed Blue/Green Deployments in Amazon Aurora and Amazon RDS
- Blue/Green deployments with AWS Elastic Beanstalk

Video correlati:

- re:Invent 2020 | Hands-off: Automating continuous delivery pipelines at Amazon
- re:Invent 2019 | Amazon's Approach to high-availability deployment

Esempi correlati:

- Prova un'implementazione blu/verde in AWS CodeDeploy
- Workshop | Creazione di pipeline CI/CD per le distribuzioni canary Lambda mediante AWS CDK
- Workshop | Creazione della prima pipeline blu/verde DevOps con Amazon ECS
- Workshop | Creazione della prima pipeline blu/verde DevOps con Amazon EKS

- Workshop | GitOps EKS con ArgoCD
- Workshop | Workshop su CI/CD su AWS
- Implementazione di CI/CD su più account con funzioni Lambda basate su container AWS SAM

OPS06-BP04 Automatizza i test e il rollback

Per aumentare la velocità, l'affidabilità e la sicurezza del processo di implementazione, rendi disponibile una strategia per le funzionalità di test e rollback automatizzate negli ambienti di pre-produzione e produzione. Automatizza i test durante l'implementazione nella produzione per simulare le interazioni umane e di sistema che verificano le modifiche implementate. Automatizza il rollback per tornare rapidamente allo stato precedente corretto noto. Il rollback deve essere avviato automaticamente in condizioni predefinite, ad esempio quando il risultato desiderato della modifica non viene raggiunto o quando il test automatico fallisce. L'automazione di queste due attività migliora la percentuale di successo delle implementazioni, riduce al minimo i tempi di ripristino e riduce il potenziale impatto sulle attività aziendali.

Risultato desiderato: i test automatici e le strategie di rollback sono integrati nella pipeline di integrazione continua e distribuzione continua (CI/CD). Il monitoraggio è in grado di eseguire la convalida in base ai criteri di successo e avviare il rollback automatico in caso di errore. Ciò riduce al minimo l'impatto sugli utenti finali e sui clienti. Ad esempio, quando tutti i risultati dei test sono stati soddisfatti, promuovi il codice nell'ambiente di produzione in cui vengono avviati i test di regressione automatizzati, sfruttando gli stessi casi di test. Se i risultati dei test di regressione non corrispondono alle aspettative, viene avviato il rollback automatico nel flusso di lavoro della pipeline.

Anti-pattern comuni:

- I tuoi sistemi non sono progettati in modo da consentire loro di essere aggiornati con release più piccole. Di conseguenza, è difficile annullare tali modifiche di massa (bulk changes) durante un'implementazione conclusasi con esito negativo.
- Il processo di implementazione consiste in una serie di passaggi manuali. Dopo aver distribuito le modifiche al carico di lavoro, inizi i test post-implementazione. Dopo il test, ti rendi conto che il tuo carico di lavoro è inutilizzabile e i clienti sono disconnessi. Inizi quindi a eseguire il rollback alla versione precedente. Tutti questi passaggi manuali ritardano il ripristino complessivo del sistema e provocano un impatto prolungato sui clienti.
- Hai impiegato del tempo a sviluppare casi di test automatizzati per funzionalità che non vengono utilizzate frequentemente nella tua applicazione, riducendo al minimo il ritorno sull'investimento nella tua capacità di eseguire test automatizzati.

- La versione è composta da applicazioni, infrastrutture, patch e aggiornamenti di configurazione indipendenti l'uno dall'altro. Tuttavia, è disponibile un'unica pipeline CI/CD che fornisce tutte le modifiche contemporaneamente. Un guasto in un componente obbliga a ripristinare tutte le modifiche, rendendo il rollback complesso e inefficiente.
- Il tuo team completa il lavoro di codifica nello sprint uno e inizia il lavoro dello sprint due, ma il tuo piano non includeva i test fino allo sprint tre. Come conseguenza, i test automatici hanno rivelato difetti dello sprint uno che dovevano essere risolti prima di poter avviare il test dei deliverable dello sprint due e l'intera release viene ritardata, rendendo inutili i test automatizzati.
- I casi di test di regressione automatizzati per la release di produzione sono completi, ma non stai monitorando lo stato del carico di lavoro. Poiché non è possibile verificare se il servizio è stato riavviato o meno, non sei sicuro se il rollback sia necessario o se sia già avvenuto.

Vantaggi dell'adozione di questa best practice: i test automatizzati aumentano la trasparenza del processo di verifica e la capacità di coprire più funzionalità in un periodo di tempo più breve. Testando e convalidando le modifiche nella produzione, è possibile identificare immediatamente i problemi. Il miglioramento della coerenza con strumenti di test automatizzati consente una migliore rilevazione dei difetti. Effettuando automaticamente il rollback alla versione precedente, l'impatto sui clienti viene ridotto al minimo. In ultima analisi, il rollback automatizzato ispira maggiore fiducia nelle capacità di implementazione riducendo l'impatto sulle attività aziendali. Nel complesso, queste funzionalità si riducono garantendo al contempo la qualità. time-to-delivery

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Automatizza i test degli ambienti implementati per verificare che i risultati siano quelli desiderati. Automatizza il rollback a uno stato corretto noto quando non vengono raggiunti i risultati previsti, per ridurre al minimo il tempo di ripristino e gli errori causati dai processi manuali. Integra gli strumenti di test con il flusso di lavoro della pipeline per testare in modo coerente e ridurre al minimo gli input manuali. Dai priorità all'automazione dei casi di test, come quelli che mitigano i rischi maggiori e devono essere testati frequentemente a ogni modifica. Inoltre, automatizza il rollback in base a condizioni specifiche predefinite nel tuo piano di test.

Passaggi dell'implementazione

1. Stabilisci un ciclo di vita di test per il tuo ciclo di vita di sviluppo che definisca ogni fase del processo di test, dalla pianificazione dei requisiti allo sviluppo dei test case, alla configurazione degli strumenti, ai test automatizzati e alla chiusura dei test case.

a. Crea un approccio di test specifico per il carico di lavoro partendo dalla tua strategia di test complessiva.

b. Prendi in considerazione una strategia di test continuo, laddove appropriato, durante tutto il ciclo di vita dello sviluppo.

2. Seleziona strumenti automatizzati per il test e il rollback in base ai requisiti aziendali e agli investimenti nella pipeline.

3. Decidi quali casi di test desideri automatizzare e quali devono essere eseguiti manualmente. Questi possono essere definiti in base alla priorità del valore aziendale della funzionalità testata. Allinea tutti i membri del team su questo piano e verifica la responsabilità per l'esecuzione di test manuali.

a. Applica le funzionalità di test automatico a casi di test specifici che è opportuno automatizzare, come i casi ripetibili o eseguiti di frequente, quelli che richiedono attività ripetitive o quelli non più necessari per più configurazioni.

b. Definisci gli script di automazione dei test e i criteri di successo nello strumento di automazione in modo da poter avviare l'automazione continua del flusso di lavoro quando casi specifici falliscono.

c. Definisci criteri di errore specifici per il rollback automatico.

4. Dai priorità all'automazione dei test per ottenere risultati coerenti con lo sviluppo accurato e completo di casi di test in cui la complessità e l'interazione umana hanno un rischio maggiore di fallimento.

5. Integra i tuoi strumenti di test e rollback automatizzati nella tua pipeline CI/CD.

a. Sviluppa criteri di successo chiari per le tue modifiche.

b. Monitora e osserva per rilevare questi criteri e annullare automaticamente le modifiche quando vengono soddisfatti criteri di rollback specifici.

6. Esegui diversi tipi di test di produzione automatizzati, come:

a. Test A/B, per mostrare i risultati rispetto alla versione corrente tra due gruppi di utenti di test.

b. Test canary, che consente di distribuire la modifica a un sottoinsieme di utenti prima di rilasciarla a tutti.

c. Test con flag delle funzionalità, che consente di attivare e disattivare una singola funzionalità della nuova versione alla volta dall'esterno dell'applicazione, in modo che ogni nuova funzionalità possa essere convalidata una alla volta.

d. Test di regressione, per verificare nuove funzionalità con componenti correlati esistenti.

7. Monitora gli aspetti operativi dell'applicazione, delle transazioni e delle interazioni con altre applicazioni e componenti. Sviluppa report per mostrare il successo delle modifiche in base al carico di lavoro in modo da poter identificare quali parti dell'automazione e del flusso di lavoro possono essere ulteriormente ottimizzate.
   a. Sviluppa report sui risultati dei test che ti aiutino a prendere decisioni rapide sull'opportunità o meno di richiamare o meno le procedure di rollback.
   b. Implementa una strategia che consenta il rollback automatico basato su condizioni di errore predefinite derivanti da uno o più metodi di test.
8. Sviluppa i tuoi casi di test automatizzati per consentire la riutilizzabilità in caso di modifiche ripetibili future.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS06-BP01 Piano per modifiche non riuscite
- OPS06-BP02 Implementazioni di test

Documenti correlati:

- AWS Builders Library | Garantire la sicurezza in caso di rollback durante le implementazioni
- Ridistribuisci e ripristina una distribuzione con AWS CodeDeploy
- 8 best practice per automatizzare le implementazioni con AWS CloudFormation

Esempi correlati:

- Test dell'interfaccia utente senza server utilizzando Selenium e Developer Tools AWS LambdaAWS FargateAWS

Video correlati:

- re:Invent 2020 | Hands-off: Automating continuous delivery pipelines at Amazon
- re:Invent 2019 | Amazon's Approach to high-availability deployment

OPS 7. Come fai a sapere se hai tutto pronto per supportare un carico di lavoro?

Valuta la prontezza operativa del carico di lavoro, dei processi e delle procedure, nonché del personale per comprendere i rischi operativi correlati al carico di lavoro.

Best practice

- OPS07-BP01 Verifica della capacità del personale
- OPS07-BP02 Revisione costante della prontezza operativa
- OPS07-BP03 Utilizzo di runbook per eseguire le procedure
- OPS07-BP04 Utilizzo dei playbook per analizzare i problemi
- OPS07-BP05 Adozione di decisioni informate per implementare sistemi e modifiche
- OPS07-BP06 Creazione dei piani di supporto per i carichi di lavoro di produzione

OPS07-BP01 Verifica della capacità del personale

Predisponi un meccanismo per stabilire se è disponibile il numero appropriato di risorse qualificate per supportare il carico di lavoro. Le risorse devono essere state formate sulla piattaforma e sui servizi che costituiscono il tuo carico di lavoro. Fornisci loro le informazioni necessarie per eseguire il carico di lavoro. Devi avere a disposizione personale qualificato sufficiente per supportare il normale funzionamento del carico di lavoro e gestire gli eventuali incidenti. Predisponi personale sufficiente per la rotazione durante la reperibilità e le ferie per evitare motivi di frustrazione.

Risultato desiderato:

- Presenza di personale qualificato sufficiente per supportare il carico di lavoro nei momenti in cui è disponibile.
- Capacità di fornire al personale formazione sul software e sui servizi che costituiscono il carico di lavoro.

Anti-pattern comuni:

- Implementazione di un carico di lavoro senza membri del team qualificati per l'esecuzione della piattaforma e dei servizi in uso.
- Mancanza di personale sufficiente per supportare la reperibilità a rotazione o le richieste di permesso del personale.

Vantaggi dell'adozione di questa best practice:

- Presenza di membri del team qualificati che offrono un supporto efficace al carico di lavoro.
- Con un numero sufficiente di membri del team, puoi supportare il carico di lavoro e la reperibilità a rotazione, riducendo il rischio di frustrazione.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Verifica che sia disponibile personale qualificato sufficiente per supportare il carico di lavoro. Assicurati che il numero di membri del team di cui disponi sia sufficiente a coprire le normali attività operative, inclusa la reperibilità a rotazione.

Esempio del cliente

AnyCompany Retail si assicura che i team che supportano il carico di lavoro includano personale qualificato sufficiente. L'azienda ha al suo interno un numero sufficiente di tecnici per supportare la reperibilità a rotazione. Il personale riceve formazione sul software e sulla piattaforma su cui è basato il carico di lavoro e viene incoraggiato a conseguire certificazioni. Vi è personale sufficiente per permettere alle persone di richiedere permessi di assenza, continuando a supportare il carico di lavoro durante la reperibilità a rotazione.

Passaggi dell'implementazione

1. Assegna un numero adeguato di risorse del personale per eseguire e supportare il carico di lavoro, tenendo conto della reperibilità, dei problemi relativi alla sicurezza e degli eventi del ciclo di vita, come la fine del supporto e le attività di rotazione dei certificati.
2. Forma il personale sul software e sulle piattaforme che costituiscono il carico di lavoro.
   a. Formazione e certificazione AWS offre una libreria di corsi su AWS. Sono disponibili corsi gratuiti e a pagamento, online e di persona.
   b. AWS organizza eventi e webinar in cui puoi apprendere dagli esperti AWS.
3. Effettua le seguenti operazioni regolarmente:
   - Valuta le dimensioni e le competenze del team in base al mutare delle condizioni operative e del carico di lavoro.
   - Adegua le dimensioni e le competenze del team ai requisiti operativi.
   - Verifica la capacità di affrontare gli eventi pianificati del ciclo di vita, la sicurezza non pianificata e le notifiche operative tramite AWS Health.

Livello di impegno per il piano di implementazione: elevato. L'assunzione e la formazione di un team per supportare il carico di lavoro possono richiedere un impegno significativo, ma assicurano solidi vantaggi a lungo termine.

Risorse

Best practice correlate:

- OPS11-BP04 Eseguire la gestione della conoscenza: i membri del team devono disporre delle informazioni necessarie per eseguire e supportare il carico di lavoro. La gestione delle informazioni è il fattore chiave a questo scopo.

Documenti correlati:

- Eventi e webinar AWS
- Formazione e certificazione AWS

OPS07-BP02 Revisione costante della prontezza operativa

Usa le revisioni sulla prontezza operativa (ORR) per verificare la possibilità di utilizzare il carico di lavoro. ORR è un meccanismo sviluppato da Amazon per verificare che i team possano utilizzare in sicurezza i propri carichi di lavoro. ORR è un processo di revisione e ispezione che utilizza un elenco di controllo per i requisiti. È un'esperienza self-service che i team utilizzano per certificare i propri carichi di lavoro. Le ORR includono le best practice delle lezioni apprese durante gli anni dedicati alla creazione di software.

Un elenco di controllo ORR è composto da suggerimenti sull'architettura, processo operativo, gestione degli eventi e qualità del rilascio. Il nostro processo di correzione dell'errore (CoE, Correction of Error) è uno dei principali fattori trainanti di questi elementi. L'analisi post-incidente deve guidare l'evoluzione della ORR. Una ORR non riguarda solo l'adozione delle best practice, ma anche la prevenzione del ripetersi di eventi già visti. Infine, in una ORR possono essere inclusi anche i requisiti di sicurezza, governance e conformità.

Esegui le ORR prima che un carico di lavoro venga lanciato nella disponibilità generale e quindi durante tutto il ciclo di vita dello sviluppo software. L'esecuzione della ORR prima del lancio aumenta la tua capacità di utilizzare il carico di lavoro in sicurezza. Riesegui periodicamente la ORR sul carico di lavoro per individuare eventuali scostamenti dalle best practice. Puoi usare gli elenchi di controllo ORR per il lancio di nuovi servizi e le ORR per le revisioni periodiche. In tal modo puoi tenerti aggiornato sulle nuove best practice che emergono e incorporare le lezioni apprese dall'analisi

post-incidente. Man mano che l'utilizzo del cloud cresce, puoi creare i requisiti di ORR nella tua architettura come valori predefiniti.

Risultato desiderato: disponi di un elenco di controllo ORR con le best practice per la tua organizzazione. Le ORR vengono eseguite prima dell'avvio dei carichi di lavoro. Le ORR vengono eseguite periodicamente nel corso del ciclo di vita del carico di lavoro.

Anti-pattern comuni:

- Avvii un carico di lavoro senza sapere se puoi utilizzarlo.
- I requisiti di governance e sicurezza non sono inclusi nella certificazione di un carico di lavoro per l'avvio.
- I carichi di lavoro non vengono rivalutati periodicamente.
- I carichi di lavoro vengono avviati senza le procedure richieste.
- Si osserva la ripetizione di errori con la stessa causa principale in più carichi di lavoro.

Vantaggi dell'adozione di questa best practice:

- I tuoi carichi di lavoro includono le best practice di architettura, processo e gestione.
- Le lezioni apprese sono incorporate nel processo ORR.
- Le procedure richieste sono in atto all'avvio dei carichi di lavoro.
- Le ORR vengono eseguite durante l'intero ciclo di vita del software dei carichi di lavoro.

Livello di rischio se questa best practice non fosse adottata: elevato

Guida all'implementazione

Una ORR è composta da un processo e un elenco di controllo. Il processo ORR deve essere adottato dall'organizzazione e supportato da uno sponsor esecutivo. Come minimo, le ORR devono essere eseguite prima che il carico di lavoro venga lanciato nella disponibilità generale. Esegui la ORR durante tutto il ciclo di vita dello sviluppo software per mantenerlo aggiornato con le best practice o i nuovi requisiti. L'elenco di controllo ORR deve includere elementi di configurazione, requisiti di sicurezza e governance e best practice dell'organizzazione. Nel tempo, puoi utilizzare servizi come, AWS Config, AWS Security Hub CSPM, e guardrail di AWS Control Tower per sviluppare best practice dall'ORR ai guardrail per il rilevamento automatico delle best practice.

Esempio del cliente

Dopo diversi incidenti di produzione, AnyCompany Retail ha deciso di implementare un processo ORR. Ha creato un elenco di controllo composto da best practice, requisiti di governance e conformità e lezioni apprese dalle interruzioni. I nuovi carichi di lavoro conducono le ORR prima dell'avvio. Ogni carico di lavoro esegue una ORR annuale con un sottoinsieme di best practice per incorporare nuove best practice e requisiti che vengono aggiunti all'elenco di controllo ORR. Nel corso del tempo, AnyCompany Retail ha utilizzato AWS Config per rilevare alcune best practice, velocizzando il processo ORR.

Passaggi dell'implementazione

Per maggiori informazioni sulle ORR, consulta il whitepaper Operational Readiness Reviews (ORR). Il documento fornisce informazioni dettagliate sulla cronologia del processo ORR, su come creare la procedura ORR e su come sviluppare il proprio elenco di controllo ORR. I passaggi seguenti costituiscono una versione abbreviata di quel documento. Per una comprensione approfondita di cosa sono le ORR e di come crearne una, ti consigliamo di leggere il whitepaper.

1. Riunisci le parti interessate importanti, inclusi i rappresentanti della sicurezza, delle operazioni e dello sviluppo.
2. Chiedi a ogni parte interessata di indicare almeno un requisito. Per la prima iterazione, prova a limitare il numero di elementi a trenta al massimo.
    - Appendix B: Example ORR questions del whitepaper Operational Readiness Reviews (ORR) contiene domande di esempio che puoi utilizzare per iniziare.
3. Raccogli i tuoi requisiti in un foglio di calcolo.
    - Puoi utilizzare gli obiettivi personalizzati di AWS Well-Architected Tool per creare la tua ORR e condividerla tra i tuoi account e la tua organizzazione AWS.
4. Identifica un carico di lavoro su cui condurre la ORR. L'ideale è un carico di lavoro pre-lancio o un carico di lavoro interno.
5. Scorri l'elenco di controllo ORR e prendi nota di tutti i rilevamenti fatti. I rilevamenti potrebbero essere validi se è in atto una mitigazione. Aggiungi qualsiasi rilevamento privo di mitigazione al tuo backlog di elementi e implementalo prima del lancio.
6. Continua ad aggiungere le best practice e i requisiti all'elenco di controllo ORR nel corso del tempo.

I clienti Supporto con supporto Enterprise possono richiedere il workshop Operational Readiness Review al proprio Technical Account Manager. Il workshop è strutturato come una sessione interattiva di lavoro a ritroso per sviluppare il proprio elenco di controllo ORR.

Livello di impegno per il piano di implementazione: elevato. L'adozione di una procedura ORR nella tua organizzazione richiede la sponsorizzazione dell'esecutivo e l'adesione delle parti interessate. Crea e aggiorna l'elenco di controllo con input provenienti da tutta l'organizzazione.

Risorse

Best practice correlate:

- OPS01-BP03 Valuta i requisiti di governance: i requisiti di governance sono una scelta naturale per un elenco di controllo ORR.
- OPS01-BP04 Valutazione dei requisiti di conformità: i requisiti di conformità sono talvolta inclusi in un elenco di controllo ORR. Altre volte costituiscono un processo separato.
- OPS03-BP07 Team di risorse appropriati: la capacità del team è un buon requisito ORR.
- OPS06-BP01 Piano per modifiche non riuscite: prima di avviare il carico di lavoro, è necessario stabilire un piano di rollback o rollforward.
- OPS07-BP01 Verifica della capacità del personale: per supportare un carico di lavoro è necessario disporre del personale necessario.
- SEC01-BP03 Identificazione e convalida degli obiettivi di controllo: gli obiettivi di controllo della sicurezza sono requisiti ORR di eccellenza.
- REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati: i piani di disaster recovery sono un buon requisito ORR.
- COST02-BP01 Sviluppo di policy basate sui requisiti dell'organizzazione: le policy di gestione dei costi sono utili da includere nell'elenco di controllo ORR.

Documenti correlati:

- AWS Control Tower: guardrail in AWS Control Tower
- AWS Well-Architected Tool - Custom Lenses
- Operational Readiness Review Template di Adrian Hornsby
- Whitepaper Operational Readiness Reviews (ORR)

Video correlati:

- Supporto AWSs You | Building an Effective Operational Readiness Review (ORR)

Esempi correlati:

- Sample Operational Readiness Review (ORR) Lens

Servizi correlati:

- AWS Config
- AWS Control Tower
- AWS Security Hub CSPM
- AWS Well-Architected Tool

OPS07-BP03 Utilizzo di runbook per eseguire le procedure

Un runbook è un processo documentato finalizzato al raggiungimento di un determinato risultato. I runbook sono composti da una serie di passaggi che è necessario eseguire per conseguire un obiettivo. L'uso dei runbook può essere fatto risalire agli albori dell'aviazione. Nelle operazioni cloud, è possibile utilizzare i runbook per ridurre i rischi e ottenere i risultati desiderati. In estrema sintesi, un runbook è un elenco di controllo da seguire per completare un'attività.

I runbook costituiscono una parte essenziale del funzionamento dei carichi di lavoro. Dall'onboarding di un nuovo membro in un team all'implementazione di una versione principale, i runbook sono processi codificati che garantiscono risultati coerenti indipendentemente da chi li utilizza. I runbook devono essere pubblicati a livello centralizzato e aggiornati in base all'evoluzione del processo. L'aggiornamento dei runbook rappresenta infatti un elemento chiave dell'intero processo di gestione delle modifiche. Devono inoltre includere le linee guida relative a gestione degli errori, strumenti, autorizzazioni, eccezioni ed escalation in caso di problemi.

Man mano che l'organizzazione cresce, è consigliabile automatizzare i runbook. Inizia con runbook concisi e di frequente utilizzo. Utilizza un linguaggio di scripting per automatizzare le procedure o semplificarne l'esecuzione. Dopo aver automatizzato i primi runbook, potrai dedicare altro tempo all'automazione dei runbook più complessi. Gradualmente dovrai automatizzare la maggior parte dei runbook.

Risultato desiderato: il team dispone di una raccolta di linee guida dettagliate per l'esecuzione delle attività relative ai carichi di lavoro. I runbook contengono il risultato desiderato, gli strumenti e le autorizzazioni necessari e le istruzioni per la gestione degli errori. Vengono archiviati in una posizione centralizzata (sistema di controllo delle versioni) e aggiornati di frequente. Ad esempio, i runbook

forniscono ai team le funzionalità per monitorare, comunicare e rispondere agli eventi AWS Health degli account critici durante gli allarmi delle applicazioni, i problemi operativi e gli eventi del ciclo di vita pianificati.

Anti-pattern comuni:

- Ricorso alla memoria per completare i singoli passaggi di un processo.
- Implementazione manuale delle modifiche senza utilizzare un elenco di controllo.
- Vari membri dei team eseguono lo stesso processo con procedure o risultati diversi.
- Mancato aggiornamento dei runbook in base alle modifiche o ai processi di automazione del sistema.

Vantaggi dell'adozione di questa best practice:

- Riduzione della percentuale degli errori per le attività manuali.
- Le operazioni vengono eseguite in modo coerente.
- I nuovi membri dei team possono essere operativi da subito.
- I runbook possono essere automatizzati per semplificare le operazioni più impegnative.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I runbook possono avere vari formati, a seconda del livello di "maturità" dell'organizzazione. Nella loro formulazione minima, devono essere un documento di testo in cui sono dettagliate le procedure. Il risultato desiderato deve essere indicato in modo chiaro e preciso. Devono inoltre documentare in modo chiaro le autorizzazioni e gli strumenti speciali necessari. Devono includere linee guida dettagliate relative alla gestione degli errori e ai livelli di escalation nel caso in cui si verifichino problemi o errori. I runbook devono riportare il nome del proprietario ed essere pubblicati in una posizione centralizzata. Dopo averlo compilato, un runbook deve essere convalidato. A tale scopo, devi far predisporre il runbook da un membro diverso del tuo team. Con l'evoluzione della procedura, aggiorna i runbook in base al processo di gestione delle modifiche.

I runbook in formato testuale devono essere automatizzati a seconda dell'evoluzione dell'organizzazione. L'utilizzo di servizi come le automazioni di AWS Systems Manager ti consentono di trasformare un testo non formattato in automazioni che possono essere eseguite nell'ambito di un carico di lavoro. Queste automazioni possono essere eseguite in risposta a eventi, per ridurre il

carico operativo a salvaguardia del carico di lavoro. AWS Systems Manager Automation offre inoltre un'esperienza di progettazione visiva a uso limitato di codice per semplificare la creazione di runbook di automazione.

Esempio del cliente

AnyCompany Retail deve eseguire aggiornamenti dello schema del database durante le implementazioni del software. Il team responsabile delle operazioni cloud ha lavorato assieme al team addetto all'amministrazione del database per redigere un runbook per l'implementazione manuale di queste modifiche. Nel runbook sono incluse le procedure dettagliate sotto forma di elenco di controllo. È presente anche una sezione sulla gestione degli errori in caso di problemi. Il runbook è stato pubblicato assieme ad altri runbook sul wiki interno. Il team responsabile delle operazioni cloud pensa di pianificare l'automazione del runbook in futuro.

Passaggi dell'implementazione

Se non è presente un repository di documenti, è consigliabile creare una libreria di runbook utilizzando un repository per il controllo delle versioni. Puoi creare i runbook utilizzando Markdown. Di seguito è riportato un modello di runbook di esempio che è possibile utilizzare come riferimento per la creazione dei runbook.

```
# Runbook Title
## Runbook Info
| Runbook ID | Description | Tools Used | Special Permissions | Runbook Author | Last
 Updated | Escalation POC |
|-------|-------|-------|-------|-------|-------|-------|
| RUN001 | What is this runbook for? What is the desired outcome? | Tools | Permissions
 | Your Name | 2022-09-21 | Escalation Name |
## Steps
1. Step one
2. Step two
```

1. Se non disponi di un repository o di un wiki per la documentazione, crea un repository per il controllo delle versioni nel sistema di controllo delle versioni in uso.
2. Individua un processo che non ha un runbook. Un processo ideale viene eseguito a cadenza più o meno regolare, con un numero limitato di passaggi e con errori a basso impatto.
3. Nel repository di documenti, crea una nuova bozza di documento Markdown utilizzando il modello. Specifica il titolo del runbook e i campi obbligatori in Informazioni runbook.
4. Partendo dal primo passaggio, compila l'area Passaggi del runbook.

5. Associa il runbook a un membro del team. Chiedi a tale membro di utilizzare il runbook per convalidare i passaggi. In caso di informazioni mancanti o poca chiarezza, aggiorna il runbook.
6. Pubblica il runbook nell'archivio della documentazione interna. Comunica l'avvenuta pubblicazione al team e alle altre parti interessate.
7. In questo modo, nel corso del tempo creerai una libreria di runbook. Man mano che la libreria cresce, comincia a pensare di automatizzare i runbook.

Livello di impegno per il piano di implementazione: basso Lo standard minimo previsto per i runbook è una guida dettagliata in formato testuale. L'automazione dei runbook può aumentare l'impegno a livello di implementazione.

Risorse

Best practice correlate:

- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure
- OPS07-BP04 Utilizzo dei playbook per analizzare i problemi
- OPS10-BP01 Utilizzo di un processo per la gestione di eventi, incidenti e problemi
- OPS10-BP02 Definizione di un processo per ogni avviso
- OPS11-BP04 Gestione delle informazioni

Documenti correlati:

- Achieving Operational Excellence using automated playbook and runbook
- AWS Systems Manager: utilizzo dei runbook
- Migration playbook for AWS large migrations - Task 4: Improving your migration runbooks
- Use AWS Systems Manager Automation runbooks to resolve operational tasks

Video correlati:

- AWS re:Invent 2019: DIY guide to runbooks, incident reports, and incident response
- How to automate IT Operations on AWS | Amazon Web Services
- Integrate Scripts into AWS Systems Manager

Esempi correlati:

- Well-Architected Labs: automazione delle operazioni con playbook e runbook
- AWS Post del blog : Build a Cloud Automation Practice for Operational Excellence: Best Practices AWS Managed Services
- AWS Systems Manager: Automation walkthroughs
- AWS Systems Manager: Restore a root volume from the latest snapshot runbook
- Building an AWS incident response runbook using Jupyter notebooks and CloudTrail Lake
- Gitlab: runbook
- Rubix: una libreria Python per la creazione di runbook in notebook Jupyter
- Using Document Builder to create a custom runbook

Servizi correlati:

- AWS Systems Manager Automation

OPS07-BP04 Utilizzo dei playbook per analizzare i problemi

I playbook sono guide dettagliate che vengono utilizzate quando si verificano incidenti per analizzare, valutare l'impatto e identificare la causa principale del problema. I playbook sono utili in molti scenari diversi, dalle implementazioni non riuscite agli incidenti di sicurezza. In molti casi, i playbook identificano la causa principale che viene poi mitigata tramite un runbook. I playbook costituiscono un componente essenziale dei piani di risposta agli incidenti di ogni organizzazione.

Un buon playbook include diverse caratteristiche principali che guidano l'utente, passo dopo passo, nel processo di rilevamento. Ma quali passaggi deve eseguire l'utente per diagnosticare un incidente? Illustra chiaramente nel playbook se sono necessari strumenti speciali o autorizzazioni elevate. È essenziale predisporre un piano di comunicazione per aggiornare le parti interessate sullo stato dell'analisi. Nelle situazioni in cui non è possibile identificare la causa principale, il playbook deve prevedere un piano di escalation. Se viene identificata la causa principale, il playbook deve includere il riferimento di un runbook che descrive come risolvere il problema. I playbook devono essere archiviati a livello centrale e aggiornati regolarmente. Se i playbook vengono utilizzati per avvisi specifici, fornisci al team i riferimenti dei playbook all'interno degli avvisi.

Man mano che l'organizzazione acquisisce maturità, puoi automatizzare i playbook. Inizia con i playbook che trattano incidenti a basso rischio. Utilizza gli script per automatizzare le procedure di rilevamento. Assicurati di avere i relativi runbook per mitigare le cause principali più comuni.

Risultato desiderato: la tua organizzazione dispone dei playbook per gli incidenti comuni. I playbook sono archiviati in una posizione centrale e disponibili per i membri del team. I playbook vengono aggiornati di frequente. Per qualsiasi causa principale nota, vengono creati i relativi runbook.

Anti-pattern comuni:

- Non esiste un modo standard per analizzare un incidente.
- I membri del team confidano nella "memoria muscolare" o nelle conoscenze istituzionali per risolvere i problemi di un'implementazione non riuscita.
- I nuovi membri del team apprendono come analizzare i problemi attraverso tentativi ed errori.
- Le best practice per l'analisi dei problemi non sono condivise tra i team.

Vantaggi dell'adozione di questa best practice:

- I playbook rendono più efficaci le tue attività di mitigazione degli incidenti.
- Uno stesso playbook può essere utilizzato da diversi membri del team in modo da identificare la causa principale in modo coerente.
- Le cause principali note possono già disporre di runbook appositamente sviluppati, accelerando i tempi di ripristino.
- I playbook contribuiscono ad accelerare la collaborazione tra i membri del team.
- I team possono applicare i processi su vasta scala tramite i playbook ripetibili.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Il modo in cui crei e utilizzi i playbook dipende dalla maturità della tua organizzazione. Se non hai familiarità con il cloud, crea i playbook in formato testo in un repository per i documenti centrale. Man mano che l'organizzazione acquisisce maturità, i playbook possono diventare semiautomatizzati tramite script scritti in linguaggi come Python. Questi script possono essere eseguiti all'interno di un notebook Jupyter per accelerare il rilevamento. Le organizzazioni avanzate dispongono di playbook completamente automatizzati per i problemi comuni che vengono risolti automaticamente con i runbook.

Inizia a creare i playbook elencando gli incidenti comuni che si verificano nel tuo carico di lavoro. Scegli i playbook per gli incidenti a basso rischio e in cui la causa principale è riconducibile a pochi

problemi. Una volta creati i playbook per gli scenari più semplici, passa agli scenari a rischio più elevato o in cui la causa principale non è ancora nota.

I playbook in formato testo vengono automatizzati man mano che l'organizzazione acquisisce maturità. L'utilizzo di servizi come le automazioni di AWS Systems Manager ti consentono di trasformare un semplice testo in automazioni eseguibili sul carico di lavoro per accelerare le analisi. Queste automazioni possono essere attivate in risposta agli eventi, riducendo il tempo medio per rilevare e risolvere gli incidenti.

Grazie a AWS Systems Manager Incident Manager, i clienti possono rispondere agli incidenti. Questo servizio fornisce un'unica interfaccia per valutare gli incidenti, informare le parti interessate circa il rilevamento e la mitigazione e collaborare per tutta la durata dell'incidente. Utilizza le automazioni di AWS Systems Manager per accelerare il rilevamento e il ripristino.

Esempio del cliente

Si è verificato un incidente che ha avuto un impatto sulla produzione dell'azienda AnyCompany Retail. L'ingegnere di turno utilizza un playbook per analizzare il problema e man mano che esegue i passaggi, mantiene aggiornate le parti interessate indicati nel playbook. L'ingegnere identifica la causa principale come una race condition di un servizio di backend. Utilizzando un runbook, l'ingegnere riavvia il servizio e riporta quindi AnyCompany Retail online.

Passaggi dell'implementazione

Se non è già presente, è consigliabile creare un repository per i documenti con il controllo delle versioni per la libreria di playbook. Puoi creare i tuoi playbook utilizzando Markdown, compatibile con la maggior parte dei sistemi di automazione dei playbook. Se parti da zero, utilizza il seguente modello di playbook come esempio.

```
# Playbook Title
## Playbook Info
| Playbook ID | Description | Tools Used | Special Permissions | Playbook Author | Last
 Updated | Escalation POC | Stakeholders | Communication Plan |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| RUN001 | What is this playbook for? What incident is it used for? | Tools |
 Permissions | Your Name | 2022-09-21 | Escalation Name | Stakeholder Name | How will
 updates be communicated during the investigation? |
## Steps
1. Step one
2. Step two
```

1. Se non disponi di un repository o di un wiki per i documenti, crea un nuovo repository di controllo per il controllo delle versioni per i tuoi playbook nel tuo sistema di controllo delle versioni.
2. Identifica un problema comune che richieda un'analisi, vale a dire uno scenario in cui la causa principale è riconducibile a pochi problemi e la risoluzione è a basso rischio.
3. Utilizzando il modello Markdown, compila la sezione Titolo del playbook e i campi in Informazioni sul playbook.
4. Includi le procedure per la risoluzione dei problemi. Illustra nel modo più chiaro possibile le azioni da eseguire o le aree da analizzare.
5. Chiedi a un membro del team di esaminare e convalidare il tuo playbook. Se manca un'informazione o è necessario un chiarimento, aggiorna il playbook.
6. Pubblica il tuo playbook nel repository per i documenti e informa il tuo team e tutte le parti interessate.
7. Questa libreria di playbook diventerà sempre più ricca man mano che ne aggiungerai altri. Una volta che sono disponibili diversi playbook, inizia ad automatizzarli con strumenti come le automazioni di AWS Systems Manager per mantenere sincronizzati l'automazione e i playbook.

Livello di impegno per il piano di implementazione: basso I playbook sono documenti di testo archiviati in una posizione centrale. Le organizzazioni che hanno acquisito maturità applicano l'automazione dei playbook.

Risorse

Best practice correlate:

- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure
- OPS07-BP03 Utilizzo di runbook per eseguire le procedure
- OPS10-BP01 Utilizzo di un processo per la gestione di eventi, incidenti e problemi
- OPS10-BP02 Definizione di un processo per ogni avviso
- OPS11-BP04 Gestione delle informazioni

Documenti correlati:

- Achieving Operational Excellence using automated playbook and runbook
- AWS Systems Manager: utilizzo dei runbook
- Use AWS Systems Manager Automation runbooks to resolve operational tasks

Video correlati:

- AWS re:Invent 2019: DIY guide to runbooks, incident reports, and incident response (SEC318-R1)
- AWS Systems Manager Incident Manager - AWS Virtual Workshops
- Integrate Scripts into AWS Systems Manager

Esempi correlati:

- AWS Framework per playbook per i clienti
- AWS Systems Manager: Automation walkthroughs
- Building an AWS incident response runbook using Jupyter notebooks and CloudTrail Lake
- Rubix: una libreria Python per la creazione di runbook in notebook Jupyter
- Using Document Builder to create a custom runbook

Servizi correlati:

- AWS Systems Manager Automation
- AWS Systems Manager Incident Manager

OPS07-BP05 Adozione di decisioni informate per implementare sistemi e modifiche

Predisponi i processi per la gestione delle modifiche al carico di lavoro che hanno restituito esito positivo e negativo. Si definisce "pre-mortem" un esercizio in cui il team simula un errore per sviluppare strategie di mitigazione. Utilizza questo esercizio per prevedere errori e creare procedure ove opportuno. Valuta vantaggi e rischi dell'implementazione di modifiche nel carico di lavoro. Verifica che tutte le modifiche siano conformi ai requisiti di governance.

Risultato desiderato:

- Adozione di decisioni informate durante l'implementazione di modifiche nel carico di lavoro.
- Modifiche conformi ai requisiti di governance.

Anti-pattern comuni:

- Implementazione di una modifica nel carico di lavoro senza un processo per la gestione di un'implementazione errata.

- Applicazione di modifiche all'ambiente di produzione che non sono conformi ai requisiti di governance.
- Implementazione di una nuova versione del carico di lavoro senza stabilire valori di riferimento per l'utilizzo delle risorse.

Vantaggi dell'adozione di questa best practice:

- L'azienda è preparata all'effetto di modifiche infruttuose al carico di lavoro.
- Le modifiche apportate al carico di lavoro sono conformi ai criteri di governance.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all’implementazione

Usa esercizi pre-mortem per sviluppare processi per la gestione di modifiche infruttuose. Documenta i processi di gestione delle modifiche infruttuose. Verifica che tutte le modifiche siano conformi ai requisiti di governance. Valuta vantaggi e rischi dell'implementazione di modifiche nel carico di lavoro.

Esempio del cliente

AnyCompany Retail svolge regolarmente esercizi pre-mortem per convalidare i propri processi di gestione delle modifiche infruttuose. L'azienda documenta i propri processi in un Wiki condiviso che aggiorna spesso. Tutte le modifiche sono conformi ai requisiti di governance.

Passaggi dell'implementazione

1. Prendi decisioni informate durante l'implementazione di modifiche nel carico di lavoro. Definisci ed esamina i criteri per un'implementazione corretta. Sviluppa scenari o criteri che avvierebbero il ripristino dello stato precedente a una modifica. Soppesa i vantaggi dell'implementazione di modifiche rispetto ai rischi di una modifica infruttuosa.
2. Verifica che tutte le modifiche siano conformi ai requisiti di governance.
3. Usa esercizi pre-mortem per pianificare la gestione delle modifiche infruttuose e documentare le strategie di mitigazione. Esegui un esercizio di simulazione di un'emergenza per modellare una modifica infruttuosa e convalidare le procedure di ripristino dello stato precedente.

Livello di impegno per il piano di implementazione: moderato L'implementazione di una procedura di pre-mortem richiede il coordinamento e l'impegno delle parti interessate in tutta l'organizzazione

Risorse

Best practice correlate:

- OPS01-BP03 Valuta i requisiti di governance: i requisiti di governance sono un fattore chiave per determinare se implementare una modifica.
- OPS06-BP01 Piano per modifiche non riuscite: predisponi piani per mitigare un'implementazione non riuscita e usa esercizi di pre-mortem per convalidarli.
- OPS06-BP02 Implementazioni di test: ogni modifica software deve essere testata nel modo adeguato prima dell'implementazione per ridurre gli errori nell'ambiente di produzione.
- OPS07-BP01 Verifica della capacità del personale: la presenza di personale qualificato sufficiente per supportare il carico di lavoro è essenziale per prendere una decisione informata riguardo all'implementazione di una modifica di sistema.

Documenti correlati:

- Amazon Web Services: rischio e conformità
- Modello di responsabilità condivisa AWS
- Governance in the Cloud AWS: The Right Balance Between Agility and Safety

OPS07-BP06 Creazione dei piani di supporto per i carichi di lavoro di produzione

Abilita il supporto per qualsiasi software e servizio a cui si affida il tuo carico di lavoro di produzione. Seleziona un livello di supporto adeguato per soddisfare le esigenze di assistenza della produzione. I piani di supporto per queste dipendenze sono necessari nel caso si verifichi un'interruzione del servizio o un problema di software. Documenta i piani di supporto e come chiedere assistenza per tutti i servizi e i fornitori di software. Implementa meccanismi di verifica per controllare che i riferimenti del supporto siano aggiornati.

Risultato desiderato:

- Implementa piani di supporto per software e servizi a cui si affidano i carichi di lavoro di produzione.
- Scegli un piano di supporto adeguato in base alle esigenze di assistenza.
- Documenta i piani e i livelli di supporto e come richiedere assistenza.

Anti-pattern comuni:

- Non hai piani di supporto per un fornitore software strategico. Il tuo carico di lavoro ne risente e non puoi fare nulla per accelerare un intervento risolutivo o per ricevere aggiornamenti tempestivi dal fornitore.
- Uno sviluppatore, che era il punto di contatto primario di un fornitore di software, ha lasciato l'azienda. Non puoi contattare direttamente l'assistenza del fornitore. Devi investire il tuo tempo per cercare le informazioni e orientarti tra sistemi di contatto generici, aumentando così il livello di impegno richiesto per intervenire quando necessario.
- Si verifica un'interruzione della produzione con un fornitore di software. Non esiste una documentazione su come inserire una richiesta di assistenza.

Vantaggi dell'adozione di questa best practice:

- Con il livello di supporto adeguato, puoi ottenere una risposta nei tempi previsti per soddisfare le esigenze in termini di livelli di servizio.
- In caso di problemi in produzione, puoi effettuare l'escalation del problema se sei un cliente assistito.
- Fornitori di software e servizi possono essere di aiuto per la risoluzione dei problemi durante un incidente.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Abilita i piani di supporto per qualsiasi fornitore di software e servizi a cui si affida il tuo carico di lavoro di produzione. Configura piani di supporto adeguati per soddisfare le esigenze di assistenza. Per i clienti AWS, questo significa abilitare il supporto Business di AWS o di livello superiore su qualsiasi account con carichi di lavoro di produzione. Incontra con regolarità i fornitori del servizio di assistenza per ricevere aggiornamenti sulle offerte di supporto, sui processi e sui contatti. Documenta come richiedere assistenza ai fornitori di software e servizi, incluso come inoltrare il problema in caso si verificasse un'interruzione. Implementa meccanismi di aggiornamento dei contatti del supporto.

Esempio del cliente

In AnyCompany Retail, tutte le dipendenze di servizi e software commerciali hanno piani di supporto. Ad esempio, l'azienda sfrutta il supporto Enterprise di AWS attivato su tutti gli account con carichi di lavoro di produzione. In caso di problemi, qualsiasi sviluppatore può inserire una richiesta di

assistenza. Esiste una pagina wiki con informazioni su come richiedere assistenza, chi contattare e quali best practice seguire per accelerare il processo di risoluzione.

Passaggi dell'implementazione

1. Collabora con le parti interessate all'interno della tua organizzazione per identificare i fornitori di software e servizi su cui si basa il tuo carico di lavoro. Documenta queste dipendenze.
2. Stabilisci le esigenze in termini di assistenza del tuo carico di lavoro. Seleziona un piano di supporto in linea con tali esigenze.
3. Per software e servizi commerciali definisci un piano di supporto con i fornitori.
    a. Sottoscrivere il supporto Business di AWS o un livello superiore per tutti gli account di produzione garantisce tempi di risposta più rapidi da Supporto AWS ed è una scelta fortemente consigliata. Se non hai il supporto premium, devi avere un piano di azione per gestire i problemi, che richiede l'aiuto di Supporto AWS. Supporto AWS offre una combinazione di strumenti e tecnologie, persone e programmi progettati per aiutarti in modo proattivo a ottimizzare le performance, ridurre i costi e innovare più rapidamente. Inoltre, AWS Business Support offre ulteriori vantaggi, tra cui l'accesso delle API a AWS Trusted Advisor e AWS Health per l'integrazione programmatica con i sistemi, oltre ad altri metodi di accesso come la Console di gestione AWS e i canali Amazon EventBridge.
4. Documenta il tuo piano di supporto nello strumento di gestione delle conoscenze. Includi come richiedere assistenza, chi avvertire se viene inviata una richiesta di assistenza e come inoltrare il problema durante un incidente. Un wiki è un buon meccanismo che consente a tutti di apportare gli aggiornamenti necessari alla documentazione, nel momento in cui vengono a conoscenza di modifiche a processi o contatti del supporto.

Livello di impegno per il piano di implementazione: basso La maggior parte di fornitori di servizi e software offre piani di supporto da attivare. Documentando e condividendo le best practice di supporto sul tuo sistema di gestione delle conoscenze, puoi verificare che il tuo team sappia cosa fare quando si verifica un problema in produzione.

Risorse

Best practice correlate:

- OPS02-BP02 Assegnazione di proprietari identificati a processi e procedure

Documenti correlati:

- Supporto AWS Plans

Servizi correlati:

- Supporto AWS Business
- Supporto AWS Enterprise

Gestione

Questions

- OPS 8. Come utilizzi l'osservabilità del carico di lavoro nella tua organizzazione?
- OPS 9. Come fai a comprendere lo stato delle operazioni?
- OPS 10. In che modo gestisci gli eventi del carico di lavoro e delle operazioni?

OPS 8. Come utilizzi l'osservabilità del carico di lavoro nella tua organizzazione?

Garantire l'integrità del carico di lavoro sfruttando l'osservabilità. Utilizzare metriche, log e tracce pertinenti per ottenere una visione completa delle prestazioni del carico di lavoro e risolvere i problemi in modo efficiente.

Best practice

- OPS08-BP01 Analizza le metriche del carico di lavoro
- OPS08-BP02 Analizza i registri dei carichi di lavoro
- OPS08-BP03 Analizza le tracce del carico di lavoro
- OPS08-BP04 Creare avvisi fruibili
- OPS08-BP05 Creare dashboard

OPS08-BP01 Analizza le metriche del carico di lavoro

Dopo aver implementato la telemetria dell'applicazione, analizza regolarmente le metriche raccolte. Sebbene latenza, richieste, errori e capacità (o quote) forniscano informazioni dettagliate sulle prestazioni del sistema, è fondamentale dare priorità alla revisione delle metriche relative ai risultati aziendali. Ciò ti assicura di prendere decisioni basate sui dati in linea con i tuoi obiettivi aziendali.

Risultato desiderato: informazioni dettagliate sulle prestazioni del carico di lavoro che guidano decisioni basate sui dati, garantendo l'allineamento con gli obiettivi aziendali.

Anti-pattern comuni:

- Analisi isolata delle metriche senza considerare il loro impatto sui risultati aziendali.
- Eccessiva dipendenza dalle metriche tecniche trascurando quelle aziendali.
- Revisione poco frequente delle metriche, perdita di opportunità di prendere decisioni in tempo reale.

Vantaggi dell'adozione di questa best practice:

- Comprensione migliorata della correlazione tra prestazioni tecniche e risultati aziendali.
- Processo decisionale migliorato basato su dati in tempo reale.
- Identificazione e mitigazione proattive dei problemi prima che influiscano sui risultati aziendali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Sfrutta strumenti come Amazon CloudWatch per eseguire analisi metriche. AWS servizi come il rilevamento delle CloudWatch anomalie e Amazon DevOps Guru possono essere utilizzati per rilevare anomalie, soprattutto quando le soglie statiche sono sconosciute o quando i modelli di comportamento sono più adatti al rilevamento delle anomalie.

Passaggi dell'implementazione

1. Analizza e revisiona: revisiona e interpreta regolarmente le metriche relative al carico di lavoro.
   a. Dai priorità alle metriche relative ai risultati aziendali rispetto a quelle puramente tecniche.
   b. Comprendi l'importanza di picchi, cali o schemi nei dati.
2. Utilizza Amazon CloudWatch: utilizza Amazon CloudWatch per una visualizzazione centralizzata e un'analisi approfondita.
   a. Configura le CloudWatch dashboard per visualizzare le tue metriche e confrontarle nel tempo.
   b. Usa i percentili CloudWatch per avere una visione chiara della distribuzione delle metriche, che può aiutarti a definire e comprendere i valori anomali. SLAs
   c. Imposta il rilevamento delle CloudWatch anomalie per identificare modelli insoliti senza fare affidamento su soglie statiche.

   d. Implementa l'osservabilità CloudWatch tra più account per monitorare e risolvere i problemi delle applicazioni che si estendono su più account all'interno di una regione.
   e. Utilizza CloudWatch Metric Insights per interrogare e analizzare i dati metrici tra account e regioni, identificando tendenze e anomalie.
   f. CloudWatch Applica Metric Math per trasformare, aggregare o eseguire calcoli sulle tue metriche per ottenere informazioni più approfondite.
3. Utilizza Amazon DevOps Guru: incorpora Amazon DevOps Guru per il suo rilevamento delle anomalie potenziato dall'apprendimento automatico per identificare i primi segnali di problemi operativi per le tue applicazioni serverless e risolverli prima che abbiano un impatto sui tuoi clienti.
4. Ottimizza in base agli approfondimenti: prendi decisioni informate sulla base dell'analisi delle metriche per adeguare e migliorare i carichi di lavoro.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS04-BP01 Identifica gli indicatori chiave di prestazione
- OPS04-BP02 Implementare la telemetria delle applicazioni

Documenti correlati:

- The Wheel Blog - Emphasizing the importance of continually reviewing metrics
- Percentile are important
- Usando AWS Cost Anomaly Detection
- CloudWatch osservabilità tra più account
- Interroga le tue metriche con Metrics Insights CloudWatch

Video correlati:

- Abilita l'osservabilità tra account in Amazon CloudWatch
- Introduzione ad Amazon DevOps Guru
- Analizza continuamente le metriche utilizzando AWS Cost Anomaly Detection

Esempi correlati:

- One Observability Workshop
- Acquisire informazioni operative AIOps con Amazon DevOps Guru

OPS08-BP02 Analizza i registri dei carichi di lavoro

L'analisi regolare dei log dei carichi di lavoro è essenziale per acquisire una comprensione più approfondita degli aspetti operativi dell'applicazione. Attraverso l'analisi, la consultazione e l'interpretazione efficiente dei dati di log, è possibile ottimizzare continuamente le prestazioni e la sicurezza delle applicazioni.

Risultato desiderato: informazioni dettagliate sul comportamento dell'applicazione e sulle operazioni derivanti da un'analisi completa dei log, che garantisce la rilevazione e la mitigazione proattiva dei problemi.

Anti-pattern comuni:

- Si trascura l'analisi dei log fino a quando non si verifica un problema critico.
- Il mancato utilizzo della suite completa degli strumenti disponibili per l'analisi dei log comporta la perdita di approfondimenti importanti.
- Si fa affidamento esclusivamente sulla revisione manuale dei log senza sfruttare le funzionalità di automazione e query.

Vantaggi dell'adozione di questa best practice:

- Identificazione proattiva dei colli di bottiglia operativi, delle minacce alla sicurezza e di altri problemi potenziali.
- Utilizzo efficiente dei dati di log per l'ottimizzazione continua dell'applicazione.
- Comprensione migliorata del comportamento dell'applicazione, facilitando il debug e la risoluzione dei problemi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Amazon CloudWatch Logs è un potente strumento per l'analisi dei log. Funzionalità integrate come CloudWatch Logs Insights e Contributor Insights rendono il processo di derivazione di informazioni significative dai log intuitivo ed efficiente.

Passaggi dell'implementazione

1. Configurazione dei CloudWatch registri: configura applicazioni e servizi per inviare i log ai registri. CloudWatch
2. Usa il rilevamento delle anomalie nei log: utilizza il rilevamento delle anomalie di Amazon CloudWatch Logs per identificare e segnalare automaticamente modelli di log insoliti. Questo strumento consente di gestire in modo proattivo le anomalie nei log e di rilevare tempestivamente i potenziali problemi.
3. Configura CloudWatch Logs Insights: usa CloudWatch Logs Insights per cercare e analizzare in modo interattivo i tuoi dati di log.
   a. Crea query per estrarre modelli, visualizzare i dati di log e ricavare approfondimenti utili.
   b. Usa l'analisi dei pattern CloudWatch di Logs Insights per analizzare e visualizzare i pattern di log frequenti. Questa funzionalità consente di comprendere le tendenze operative più comuni e i potenziali valori anomali nei dati di log.
   c. Usa CloudWatch Logs compare (diff) per eseguire analisi differenziali tra diversi periodi di tempo o tra diversi gruppi di log. Questa funzionalità ti consente di individuare le modifiche e valutarne l'impatto sulle prestazioni o sul comportamento del sistema.
4. Monitora i log in tempo reale con Live Tail: usa Amazon CloudWatch Logs Live Tail per visualizzare i dati dei log in tempo reale. Puoi monitorare attivamente le attività operative dell'applicazione man mano che si verificano, ottenendo una visibilità immediata sulle prestazioni del sistema e sui potenziali problemi.
5. Sfrutta Contributor Insights: utilizza CloudWatchContributor Insights per identificare i migliori oratori in dimensioni ad alta cardinalità come gli indirizzi IP o gli user-agent.
6. Implementa i filtri metrici CloudWatch Logs: configura i filtri metrici CloudWatch Logs per convertire i dati di log in metriche utilizzabili. In questo modo puoi impostare allarmi o analizzare ulteriormente i modelli.
7. Implementa l'osservabilità CloudWatch tra account: monitora e risolvi i problemi delle applicazioni che si estendono su più account all'interno di una regione.

8. Rivedi regolarmente e perfeziona: rivedi periodicamente le tue strategie di analisi dei log per acquisire tutte le informazioni pertinenti e ottimizzare continuamente le prestazioni delle applicazioni.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS04-BP01 Identifica gli indicatori chiave di prestazione
- OPS04-BP02 Implementare la telemetria delle applicazioni
- OPS08-BP01 Analizza le metriche del carico di lavoro

Documenti correlati:

- Analisi dei dati di registro con Logs Insights CloudWatch
- Utilizzo di Contributor Insights CloudWatch
- Creazione e gestione di filtri CloudWatch metrici di log

Video correlati:

- Analizza i dati di log con CloudWatch Logs Insights
- Usa CloudWatch Contributor Insights per analizzare dati ad alta cardinalità

Esempi correlati:

- CloudWatch Registra interrogazioni di esempio
- One Observability Workshop

OPS08-BP03 Analizza le tracce del carico di lavoro

L'analisi dei dati di tracciamento è fondamentale per ottenere una visione completa del percorso operativo di un'applicazione. Visualizzando e comprendendo le interazioni tra i vari componenti, consente di ottimizzare le prestazioni, identificare i colli di bottiglia e migliorare l'esperienza utente.

Risultato desiderato: ottieni una chiara visibilità sulle operazioni distribuite della tua applicazione, che si traduce in una risoluzione più rapida dei problemi e in un'esperienza utente migliorata.

Anti-pattern comuni:

- I dati di tracciamento vengono trascurati e ci si affida esclusivamente a log e metriche.
- I dati di tracciamento non sono correlati ai log associati.
- Vengono ignorate le metriche derivate dalle tracce, come la latenza e i tassi di errore.

Vantaggi dell'adozione di questa best practice:

- Migliora la risoluzione dei problemi e riduci il tempo medio di risoluzione (). MTTR
- Informazioni dettagliate sulle dipendenze e sul loro impatto.
- Identificazione e correzione rapide dei problemi legati alle prestazioni.
- Vengono sfruttate le metriche derivate dalle tracce per un processo decisionale informato.
- Esperienze utente migliorate attraverso interazioni con i componenti ottimizzate.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

AWS X-Ray offre una suite completa per l'analisi dei dati di tracciamento, fornendo una visione olistica delle interazioni con i servizi, monitorando le attività degli utenti e rilevando i problemi di prestazioni. Funzionalità come X-Ray Insights ServiceLens, X-Ray Analytics e Amazon DevOps Guru migliorano la profondità delle informazioni fruibili derivate dai dati di tracciamento.

Passaggi dell'implementazione

I seguenti passaggi offrono un approccio strutturato per implementare efficacemente l'analisi dei dati di traccia utilizzando i servizi: AWS

1. Integrazione AWS X-Ray: assicurati che X-Ray sia integrato con le tue applicazioni per acquisire dati di traccia.
2. Analizza le metriche di X-Ray: approfondisci le metriche ottenute dalle tracce di X-Ray, come latenza, tassi di richieste, tassi di errore e distribuzioni dei tempi di risposta, utilizzando la mappa dei servizi per il monitoraggio dello stato delle applicazioni.

3. Utilizzo ServiceLens: sfrutta la ServiceLensmappa per una migliore osservabilità dei tuoi servizi e delle tue applicazioni. Fornisce la visualizzazione integrata di tracce, metriche, log, allarmi e altre informazioni correlate all'integrità.
4. Abilita X-Ray Insights:
   a. Attiva X-Ray Insights per rilevare in automatico le anomalie nelle tracce.
   b. Esamina gli approfondimenti per individuare i modelli e determinare le cause ultime, come l'aumento dei tassi di errore o delle latenze.
   c. Consulta la cronologia degli approfondimenti per un'analisi cronologica dei problemi rilevati.
5. Usa X-Ray Analytics: X-Ray Analytics ti consente di approfondire i dati di tracciamento, individuare modelli ed estrarre informazioni dettagliate.
6. Usa i gruppi di X-Ray: crea gruppi in X-Ray per filtrare le tracce in base a criteri come l'elevata latenza, per un'analisi più mirata.
7. Incorpora Amazon DevOps Guru: coinvolgi Amazon DevOps Guru per trarre vantaggio dai modelli di apprendimento automatico che individuano le anomalie operative nelle tracce.
8. Usa CloudWatch Synthetics: Usa Synthetics per creare CloudWatchcanarie per il monitoraggio continuo degli endpoint e dei flussi di lavoro. Questi canary possono integrarsi con X-Ray per fornire dati di tracciamento per un'analisi approfondita delle applicazioni testate.
9. Usa Real User Monitoring (RUM): con AWS X-Ray and CloudWatch RUM, puoi analizzare ed eseguire il debug del percorso della richiesta partendo dagli utenti finali della tua applicazione fino ai servizi gestiti a valle. AWS In questo modo, puoi identificare le tendenze e gli errori di latenza che hanno un impatto sugli utenti finali.
10. Effettua le correlazioni con i log: correla i dati di tracciamento con i log correlati all'interno della relativa vista di X-Ray per una prospettiva granulare sul comportamento delle applicazioni. Ciò consente di visualizzare gli eventi del log associati direttamente alle transazioni tracciate.
11. Implementa l'osservabilità CloudWatch tra account: monitora e risolvi i problemi delle applicazioni che si estendono su più account all'interno di una regione.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS08-BP01 Analizza le metriche del carico di lavoro
- OPS08-BP02 Analizza i registri dei carichi di lavoro

Documenti correlati:

- Utilizzo ServiceLens per monitorare l'integrità delle applicazioni
- Esplorazione dei dati delle tracce con X-Ray Analytics
- Individuazione delle anomalie nelle tracce con X-Ray Insights
- Monitoraggio continuo con CloudWatch Synthetics

Video correlati:

- Analizza ed esegui il debug di applicazioni con Amazon CloudWatch Synthetics & AWS X-Ray
- Use AWS X-Ray Insights

Esempi correlati:

- One Observability Workshop
- Implementazione di X-Ray con AWS Lambda
- CloudWatchModelli Synthetics Canary

OPS08-BP04 Creare avvisi fruibili

Rilevare e rispondere tempestivamente alle deviazioni di comportamento dell'applicazione è fondamentale. È importante riconoscere quando i risultati basati sugli indicatori chiave di prestazione (KPI) sono a rischio o quando si verificano anomalie impreviste. Basare gli avvisi sui KPI garantisce che i segnali ricevuti siano direttamente correlati all'impatto aziendale od operativo. Questo approccio verso avvisi fruibili promuove risposte proattive e aiuta a mantenere le prestazioni e l'affidabilità del sistema.

Risultati desiderati: si ricevono avvisi tempestivi, pertinenti e fruibili per l'identificazione e la mitigazione rapida di potenziali problemi, soprattutto quando i risultati dei KPI sono a rischio.

Anti-pattern comuni:

- Si impostano troppi avvisi non critici, con conseguente affaticamento da avvisi ("alert fatigue").
- Non viene data priorità agli avvisi in base ai KPI, il che rende difficile comprendere l'impatto dei problemi sull'azienda.
- Non si affrontano le cause principali porta a ricevere avvisi ripetuti per lo stesso problema.

Vantaggi dell'adozione di questa best practice:

- Riduzione dell'affaticamento da avvisi ("alert fatigue") concentrandosi su avvisi pertinenti e fruibili.
- Maggiore operatività e affidabilità del sistema grazie al rilevamento e alla mitigazione proattiva dei problemi.
- Migliore collaborazione tra team e risoluzione più rapida dei problemi grazie all'integrazione con i più diffusi strumenti di avviso e comunicazione.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per creare un meccanismo di avviso efficace, è fondamentale utilizzare metriche, log e dati di tracciamento che segnalino quando i risultati basati sui KPI sono a rischio o vengono rilevate anomalie.

Passaggi dell'implementazione

1. Determina gli indicatori chiave di prestazione (KPI):: identifica gli indicatori chiave di prestazione (KPI) dell'applicazione. Gli avvisi devono essere correlati a questi KPI per riflettere accuratamente l'impatto aziendale.
2. Implementa il rilevamento delle anomalie:
   - Usa il rilevamento delle anomalie di Amazon CloudWatch: configura il rilevamento delle anomalie di Amazon CloudWatch in modo da rilevare in automatico modelli insoliti, così da generare avvisi solo per anomalie reali.
   - Utilizza AWS X-Ray Insights:
     a. Configura X-Ray Insights per la rilevazione delle anomalie nei dati di tracciamento.
     b. Configura le notifiche per X-Ray Insights così da ricevere avvisi sui problemi rilevati.
   - Esegui l'integrazione con Amazon DevOps Guru:
     a. Sfrutta Amazon DevOps Guru e le sue capacità di machine learning nel rilevare anomalie operative con i dati esistenti.
     b. Accedi alle impostazioni di notifica in DevOps Guru per la configurazione degli avvisi per le anomalie.
3. Implementa avvisi fruibili: progetta avvisi che forniscano informazioni adeguate per intraprendere un'azione immediata.

1. Monitora gli eventi AWS Health con le regole di Amazon EventBridge o effettua l'integrazione a livello di programmazione dell'API AWS Health per automatizzare le azioni in caso di ricezione di eventi AWS Health. Può trattarsi di azioni generali, come l'invio di tutti i messaggi pianificati sugli eventi del ciclo di vita a un'interfaccia di chat, oppure azioni specifiche, come l'avvio di un flusso di lavoro in uno strumento di gestione dei servizi IT.

4. Riduci l'affaticamento dagli avvisi: riduci al minimo gli avvisi non critici. Quando i team sono sovraccaricati da numerosi avvisi insignificanti, possono trascurare i problemi critici, riducendo l'efficacia complessiva del meccanismo di avviso.
5. Configura allarmi compositi: utilizza gli allarmi compositi di Amazon CloudWatch per consolidare più allarmi.
6. Integra strumenti per gli avvisi: inserisci strumenti come Ops Genie e PagerDuty.
7. Impiega Amazon Q Developer nelle applicazioni di chat: Integra Amazon Q Developer nelle applicazioni di chat per inoltrare avvisi ad Amazon Chime, Microsoft Teams e Slack.
8. Usa gli avvisi basati sui log: utilizza i filtri delle metriche dei log in CloudWatch per creare allarmi basati su eventi del log specifici.
9. Rivedi e itera: riesamina e ottimizza regolarmente le configurazioni degli avvisi.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS04-BP01 Identifica gli indicatori chiave di prestazione
- OPS04-BP02 Implementare la telemetria delle applicazioni
- OPS04-BP03 Implementare la telemetria dell'esperienza utente
- OPS04-BP04 Implementazione della telemetria delle dipendenze
- OPS04-BP05 Implementare la tracciabilità distribuita
- OPS08-BP01 Analizza le metriche del carico di lavoro
- OPS08-BP02 Analizza i registri dei carichi di lavoro
- OPS08-BP03 Analizza le tracce del carico di lavoro

Documenti correlati:

- Using Amazon CloudWatch alarms
- Create a composite alarm
- Create a CloudWatch alarm based on anomaly detection
- Notifiche DevOps Guru
- Notifiche X-Ray Insights
- Monitora, gestisci e risolvi i problemi delle tue risorse AWS con ChatOps interattive
- Amazon CloudWatch Integration Guide | PagerDuty
- Integrate Opsgenie with Amazon CloudWatch

Video correlati:

- Create Composite Alarms in Amazon CloudWatch
- Panoramica di Amazon Q Developer nelle applicazioni di chat
- AWS On Air ft. Mutative Commands in Amazon Q Developer nelle applicazioni di chat

Esempi correlati:

- Alarms, incident management, and remediation in the cloud with Amazon CloudWatch
- Tutorial: Creating an Amazon EventBridge rule that sends notifications to Amazon Q Developer in chat applications
- One Observability Workshop

OPS08-BP05 Creare dashboard

Le dashboard rappresentano la visualizzazione incentrata sull'utente dei dati di telemetria dei carichi di lavoro. Sebbene forniscano un'interfaccia visiva fondamentale, non dovrebbero sostituire i meccanismi di allarme, ma integrarli. Se realizzate con cura, sono in grado di fornire approfondimenti rapidi sullo stato e sulle prestazioni del sistema e possono informare le parti interessate in tempo reale riguardo ai risultati aziendali e all'impatto dei problemi.

Risultato desiderato:

Approfondimenti chiari e fruibili sullo stato del sistema e dell'azienda attraverso rappresentazioni visive.

Anti-pattern comuni:

- Dashboard eccessivamente complicate con troppe metriche.
- Affidarsi a dashboard senza avvisi per il rilevamento delle anomalie.
- Non aggiornare le dashboard man mano che i carichi di lavoro si evolvono.

Vantaggi di questa best practice:

- Visibilità immediata delle metriche e dei KPI critici di sistema.
- Miglioramento della comunicazione e della comprensione con le parti interessate.
- Approfondimenti rapidi sull'impatto dei problemi operativi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Dashboard incentrate sull'azienda

Le dashboard personalizzate in base ai KPI aziendali coinvolgono una gamma più ampia di parti interessate. Anche se queste persone potrebbero non essere interessate alle metriche di sistema, desiderano comprendere le implicazioni aziendali di questi numeri. Una dashboard incentrata sull'azienda garantisce che tutte le metriche tecniche e operative monitorate e analizzate siano allineate con gli obiettivi aziendali generali. Questo allineamento fornisce chiarezza, garantendo che tutti siano sulla stessa lunghezza d'onda per quanto riguarda ciò che è essenziale e ciò che non lo è. Inoltre, le dashboard che mettono in evidenza i KPI aziendali tendono ad essere più fruibili. Le parti interessate possono comprendere rapidamente lo stato delle operazioni, le aree che richiedono attenzione e il potenziale impatto sui risultati aziendali.

Con questo in mente, al momento di creare una dashboard, assicurati che ci sia un equilibrio tra metriche tecniche e KPI aziendali. Entrambi sono fondamentali, ma si rivolgono a un pubblico diverso. Idealmente, dovresti disporre di dashboard che forniscano una visione olistica dello stato e delle prestazioni del sistema, mettendo in evidenza al contempo i principali risultati aziendali e le loro implicazioni.

Le dashboard di Amazon CloudWatch sono home page personalizzabili nella console CloudWatch che è possibile usare per monitorare le risorse in un'unica vista, anche quando le risorse si trovano in vari account e regioni Regioni AWS.

Passaggi dell'implementazione

1. Crea una dashboard di base: crea una nuova dashboard in CloudWatch, assegnandole un nome esplicativo.
2. Usa i widget Markdown: prima di utilizzare le metriche, usa i widget Markdown per aggiungere un contesto testuale nella parte superiore della tua dashboard. Questo contesto specifica cosa include la dashboard, qual è l'importanza delle metriche rappresentate e può contenere anche link ad altre dashboard e strumenti di risoluzione dei problemi.
3. Crea le variabili della dashboard: integra le variabili della dashboard, se necessario, in modo da offrire visualizzazioni dinamiche e flessibili della dashboard.
4. Crea i widget per le metriche: aggiungi i widget per le metriche in modo da visualizzare varie metriche emesse dall'applicazione e personalizza questi widget in modo che rappresentino efficacemente lo stato del sistema e i risultati aziendali.
5. Esegui query con Log Insights: utilizza Approfondimenti di CloudWatch Logs per ottenere metriche fruibili dai log e visualizzare tali informazioni sulla dashboard.
6. Configura gli allarmi: integra gli allarmi CloudWatch nella dashboard per una rapida visualizzazione di tutte le metriche che violano le relative soglie.
7. Usa Contributor Insights: integra CloudWatch Contributor Insights per analizzare i campi ad alta cardinalità e comprendere meglio i principali collaboratori della tua risorsa.
8. Progetta widget personalizzati: per esigenze specifiche non soddisfatte dai widget standard, prendi in considerazione la creazione di widget personalizzati, che possono attingere da varie origini dati o rappresentare i dati in modi unici.
9. Usa AWS Health: AWS Health è la fonte autorevole di informazioni sull'integrità delle risorse Cloud AWS. Usa subito Dashboard AWS Health o usa i dati di AWS Health nei pannelli di controllo e negli strumenti in modo da avere a disposizione le informazioni giuste per prendere decisioni informate.
10. Itera e perfeziona: man mano che la tua applicazione si evolve, riesamina regolarmente la dashboard per assicurarne la pertinenza.

Risorse

Best practice correlate:

- OPS04-BP01 Identifica gli indicatori chiave di prestazione
- OPS08-BP01 Analizza le metriche del carico di lavoro

- OPS08-BP02 Analizza i registri dei carichi di lavoro
- OPS08-BP03 Analizza le tracce del carico di lavoro
- OPS08-BP04 Creare avvisi fruibili

Documenti correlati:

- Creazione di pannelli di controllo per visibilità operativa
- Using Amazon CloudWatch Dashboards

Video correlati:

- Create Cross Account & Cross Region CloudWatch Dashboards
- AWS re:Invent 2021 - Gain enterprise visibility with Cloud AWS operation dashboards)

Esempi correlati:

- One Observability Workshop
- Monitoraggio delle applicazioni con Amazon CloudWatch
- AWS Health Events Intelligence Dashboards and Insights
- Visualize AWS Health events using Amazon Managed Grafana

OPS 9. Come fai a comprendere lo stato delle operazioni?

Definisci, acquisisci e analizza i parametri delle operazioni per ottenere visibilità sugli eventi delle operazioni, in modo da intraprendere le azioni appropriate.

Best practice

- OPS09-BP01 Misura gli obiettivi operativi e i KPI con le metriche
- OPS09-BP02 Comunicare lo stato e le tendenze per garantire la visibilità delle operazioni
- OPS09-BP03 Revisione delle metriche operative e assegnazione delle priorità per favorire il miglioramento

OPS09-BP01 Misura gli obiettivi operativi e i KPI con le metriche

Ottieni obiettivi e KPI dalla tua organizzazione che definiscano il successo delle operazioni e stabilisci metriche che li riflettano. Definisci previsioni da utilizzare come riferimento e rivalutale regolarmente. Sviluppa meccanismi per raccogliere queste metriche dai team per la valutazione. Le metriche DevOps Research and Assessment (DORA) forniscono un metodo popolare per misurare i progressi verso le procedure DevOps di distribuzione del software.

Risultato desiderato:

- L'organizzazione pubblica e condivide gli obiettivi e i KPI per i team operativi.
- Stabilisci metriche che riflettono questi KPI. Gli esempi possono includere:
  - Lunghezza della coda dei ticket o età media del ticket
  - Numero di ticket raggruppati per tipo di problema
  - Tempo impiegato per lavorare ai problemi con o senza una procedura operativa standardizzata (SOP)
  - Tempo impiegato per il ripristino dopo un push di codice non riuscito
  - Volume delle chiamate

Anti-pattern comuni:

- Le scadenze di implementazione non vengono rispettate perché gli sviluppatori sono costretti a dedicarsi alle attività di risoluzione dei problemi. I team di sviluppo chiedono più personale, ma non possono quantificarne il numero perché il tempo impiegato non può essere misurato.
- È stato installato un desk di livello 1 per gestire le chiamate degli utenti. Nel corso del tempo, sono aumentati i carichi di lavoro ma non il personale assegnato al desk di livello 1. La soddisfazione dei clienti ne risente a causa dell'aumento dei tempi di chiamata e di quelli per arrivare a una soluzione, ma la dirigenza non vede indicatori di questo problema e non intraprende azioni.
- Un carico di lavoro problematico è stato affidato a un team operativo separato per la gestione. A differenza di altri carichi di lavoro, questo non è accompagnato dalla documentazione e dai runbook adeguati. Pertanto, i team dedicano più tempo alla risoluzione dei problemi e alla gestione degli errori. Tuttavia, non esistono metriche che lo documentino, il che rende difficile comprendere le responsabilità.

Vantaggi dell'adozione di questa best practice: quando il monitoraggio del carico di lavoro mostra lo stato delle nostre applicazioni e servizi, i team operativi dedicati al monitoraggio forniscono ai

proprietari informazioni dettagliate sui cambiamenti avvenuti tra i consumatori di tali carichi di lavoro, come le mutate esigenze aziendali. Misura l'efficacia di questi team e valutali rispetto agli obiettivi aziendali creando metriche in grado di riflettere lo stato delle operazioni. Le metriche possono evidenziare problemi relativi al supporto o identificare quando si verificano deviazioni rispetto a un obiettivo di livello di servizio.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Fissa un appuntamento con i leader aziendali e le parti interessate per stabilire quali saranno gli obiettivi generali del servizio. Stabilisci quali devono essere i compiti dei vari team operativi e quali sfide potrebbero affrontare. Utilizza queste informazioni per un'attività di brainstorming sugli indicatori chiave di prestazione (KPI) che potrebbero riflettere questi obiettivi operativi. Questi potrebbero essere la soddisfazione del cliente, il tempo trascorso dall'ideazione della funzionalità alla sua implementazione, il tempo medio di risoluzione dei problemi o l'efficienza in termini di costi.

Partendo dai KPI, identifica le metriche e le origini di dati che potrebbero rispecchiare al meglio questi obiettivi. La soddisfazione del cliente può essere una combinazione di diverse metriche, come i tempi di attesa o di risposta durante le chiamate, i punteggi di soddisfazione e i tipi di problemi sollevati. I tempi di implementazione possono essere la somma del tempo necessario per il test e l'implementazione, con l'aggiunta di eventuali correzioni post-implementazione. Le statistiche che mostrano il tempo dedicato a diversi tipi di problemi (o il numero di tali problemi) possono fornire indicazioni su dove è necessario un impegno mirato.

Risorse

Documenti correlati:

- Quick: utilizzo dei KPI
- Amazon CloudWatch - Using Metrics
- Creazione di pannelli di controllo
- How to track your cost optimization KPIs with KPI Dashboard
- AWS DevOps Guidance

Esempi correlati:

- Monitor the performance of your software delivery using native AWS monitoring and observability tools

- Balance deployment speed and stability with DORA metrics
- Example MLOps operational metrics in the financial services industry
- How to track your cost optimization KPIs with the KPI Dashboard

OPS09-BP02 Comunicare lo stato e le tendenze per garantire la visibilità delle operazioni

Conoscere lo stato delle operazioni e la direzione verso la quale tendono a muoversi è necessario per identificare quando i risultati possono essere a rischio, se è possibile supportare o meno carichi di lavoro aggiuntivi o per verificare gli effetti che le modifiche hanno avuto sui team. Durante gli eventi operativi, disporre di pagine di stato a cui gli utenti e i team operativi possono fare riferimento per ottenere informazioni può ridurre la pressione sui canali di comunicazione e diffondere informazioni in modo proattivo.

Risultato desiderato:

- I responsabili delle operazioni hanno a disposizione informazioni dettagliate per conoscere il volume di chiamate che i loro team stanno gestendo e quali operazioni sono in corso, ad esempio le implementazioni.
- Quando si verificano eventi che possono compromettere le normali operazioni, vengono inviati avvisi alle parti interessate e alle comunità di utenti.
- Quando ricevono un avviso o si verifica un problema, la leadership dell'organizzazione e le parti interessate possono controllare una pagina di stato e ottenere informazioni relative a un evento operativo, come punti di contatto, informazioni sui ticket e tempi di ripristino stimati.
- I report messi a disposizione della leadership e delle parti interessate contengono statistiche operative come il volume delle chiamate in un periodo di tempo, i punteggi di soddisfazione degli utenti, il numero e l'età di ticket in sospeso.

Anti-pattern comuni:

- Se un carico di lavoro si interrompe, il servizio diventa non disponibile. Il volume delle chiamate aumenta quando gli utenti chiedono di sapere cosa sta succedendo. Le richieste dei manager di sapere chi sta risolvendo un problema comportano un ulteriore aumento del volume. Vari team operativi duplicano gli sforzi mentre effettuano indagini.
- La volontà di acquisire una nuova capacità porta a riassegnare gli sforzi di alcuni membri del personale verso compiti di tipo tecnico. Non viene fornito alcun backfill e i tempi di risoluzione dei problemi aumentano. Queste informazioni non vengono acquisite e i manager vengono

a conoscenza del problema solo dopo diverse settimane o quando viene ricevuto il feedback negativo degli utenti.

Vantaggi dell'adozione di questa best practice: a volte, durante eventi operativi che hanno un impatto sull'azienda, si spreca molto tempo ed energia in query per ottenere informazioni da vari team nel tentativo di comprendere la situazione. Grazie alla creazione di pagine di stato e dashboard ampiamente diffuse, le parti interessate possono ottenere rapidamente informazioni, ad esempio, se è stato rilevato o meno un problema, chi è a capo delle attività di risoluzione o quando è previsto un ritorno alle normali operazioni. Ciò permette ai membri del team di avere più tempo per affrontare i problemi, perché non devono dilungarsi a comunicare lo stato agli altri.

Inoltre, pannelli di controllo e report forniscono informazioni ai responsabili delle decisioni e alle parti interessate in modo da scoprire se i team operativi sono in grado di rispondere alle esigenze aziendali e le modalità di allocazione delle relative risorse. Questo aspetto è fondamentale per determinare la presenza di risorse adeguate a supporto dell'azienda.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Crea pannelli di controllo che mostrino le metriche fondamentali attuali per i tuoi team operativi e rendile facilmente accessibili ai responsabili operativi e ai manager.

Crea pagine di stato aggiornabili rapidamente per diffondere informazioni relative a un incidente o un evento, come chi ne è responsabile e chi coordina la risposta. Condividi in questa pagina eventuali passaggi o soluzioni alternative che gli utenti dovrebbero prendere in considerazione e divulga ampiamente la posizione della pagina. Incoraggia gli utenti a controllare prima questa pagina quando si trovano di fronte a un problema sconosciuto.

Raccogli e fornisci report che mostrino le condizioni delle operazioni nel tempo e distribuiscili a leader e responsabili decisionali per illustrare il lavoro dei team operativi e le loro sfide ed esigenze.

Condividi con i team le metriche e i report che meglio riflettono gli obiettivi e i KPI e come hanno influito nel guidare il cambiamento. Dedica del tempo a queste attività per aumentare l'importanza delle operazioni nei e tra i team.

Usa AWS Health insieme ai pannelli di controllo o integra gli eventi AWS Health in essi, in modo che i team possano correlare i problemi relativi alle applicazioni allo stato del servizio AWS.

Risorse

Best practice correlate:

- OPS09-BP01 Misura gli obiettivi operativi e i KPI con le metriche

Documenti correlati:

- Measure Progress
- Creazione di pannelli di controllo per visibilità operativa

Esempi correlati:

- Data Operations
- How to track your cost optimization KPIs with KPI Dashboard
- The Importance of Key Performance Indicators (KPIs) for Large-Scale Cloud Migrations

OPS09-BP03 Revisione delle metriche operative e assegnazione delle priorità per favorire il miglioramento

L'assegnazione di tempo e risorse per la revisione dello stato delle operazioni garantisce che servire il settore d'attività rimanga una priorità quotidiana. Effettua regolarmente riunioni con i responsabili operativi e le parti interessate per rivedere le metriche, riconfermare o modificare traguardi e obiettivi e dare priorità ai miglioramenti.

Risultato desiderato:

- I responsabili operativi e il personale si incontrano regolarmente per esaminare le metriche in un determinato periodo di riferimento. Si comunicano le sfide, si celebrano le vittorie e si condividono le lezioni apprese.
- Parti interessate e leader aziendali vengono regolarmente informati sullo stato delle operazioni e sollecitati a fornire input su obiettivi, KPI e iniziative future. Vengono discusse e contestualizzate le scelte tra erogazione dei servizi, operazioni e manutenzione.

Anti-pattern comuni:

- Viene lanciato un nuovo prodotto, ma i team operativi di livello 1 e 2 non sono adeguatamente formati per fornire supporto oppure non dispongono di personale aggiuntivo. I leader non vedono le metriche che mostrano la diminuzione dei tempi di risoluzione dei ticket e l'aumento del volume degli incidenti. Si agisce settimane dopo, quando i numeri delle sottoscrizioni iniziano a diminuire a causa di utenti scontenti che abbandonano la piattaforma.
- Da molto tempo esiste un processo manuale per eseguire la manutenzione su un carico di lavoro. La volontà di automatizzare, seppur presente, costituiva una priorità bassa data la scarsa importanza del sistema. Nel corso del tempo, tuttavia, l'importanza del sistema è cresciuta e ora i team operativi sono impegnati per la maggior parte del tempo in questi processi manuali. Non sono previste risorse per fornire una maggiore strumentazione ai team operativi oberati dall'aumento dei carichi di lavoro, con rischi di burnout per il personale. La leadership viene a conoscenza del problema una volta segnalato da un membro del personale che lascia l'azienda per un concorrente.

Vantaggi dell'adozione di questa best practice: in alcune organizzazioni, può diventare difficile dedicare lo stesso tempo e la stessa attenzione alla fornitura di servizi e a nuovi prodotti od offerte. Quando ciò si verifica, il settore d'attività può risentirne a causa del lento deterioramento del livello di servizio atteso. Questo perché le operazioni non cambiano e non si evolvono di pari passo con la crescita del business e possono diventare presto obsolete. Senza una revisione regolare delle informazioni raccolte dai team operativi, il rischio che l'azienda corre potrebbe diventare visibile solo quando è troppo tardi. Dedicare tempo alla revisione delle metriche e delle procedure insieme al personale operativo e alla leadership, permette di mettere in luce il ruolo cruciale svolto dai team operativi nell'identificare i rischi molto prima che raggiungano livelli critici. I team operativi ottengono una visione migliore dei cambiamenti e delle iniziative aziendali imminenti, il che permette di intraprendere azioni proattive. Grazie alla visibilità delle metriche operative, la leadership è consapevole del ruolo che i team operativi svolgono nel garantire la soddisfazione dei clienti, sia interni che esterni, ed è in grado di valutare meglio le scelte in base alle priorità o di garantire che ci sia sufficiente tempo per modificare e fare evolvere operazioni e risorse attraverso nuove iniziative aziendali e di carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Dedica del tempo alla revisione delle metriche operative con le parti interessate e i team operativi e alla revisione dei dati dei report. Inserisci questi report nel contesto degli scopi e degli obiettivi

dell'organizzazione per stabilire se vengono raggiunti. Individua le cause di ambiguità in caso di obiettivi non chiari o potenziali conflitti tra quanto richiesto e quanto offerto.

Identifica come il tempo, le persone e gli strumenti possono contribuire agli esiti delle operazioni. Stabilisci quali KPI ne verrebbero influenzati e quali devono essere gli obiettivi di successo. Effettua regolarmente una revisione per assicurarti che i team operativi dispongano di risorse sufficienti per supportare il settore d'attività.

Risorse

Documenti correlati:

- Amazon Athena
- Documentazione di riferimento su parametri e dimensioni di Amazon CloudWatch
- Amazon Quick
- AWS Glue
- AWS Glue Data Catalog
- Raccolta di parametri e log da istanze Amazon EC2 e da server on-premises con l'agente Amazon CloudWatch
- Using Amazon CloudWatch metrics

OPS 10. In che modo gestisci gli eventi del carico di lavoro e delle operazioni?

Prepara e convalida le procedure in risposta agli eventi per ridurre al minimo il loro impatto sul tuo carico di lavoro.

Best practice

- OPS10-BP01 Utilizzo di un processo per la gestione di eventi, incidenti e problemi
- OPS10-BP02 Definizione di un processo per ogni avviso
- OPS10-BP03 Definizione della priorità degli eventi operativi in base agli effetti sul business
- OPS10-BP04 Definizione dei percorsi di escalation
- OPS10-BP05 Definizione di un piano di comunicazione con i clienti per eventi che incidono sul servizio
- OPS10-BP06 Comunicazione dello stato tramite pannelli di controllo

- OPS10-BP07 Automatizza le risposte agli eventi

OPS10-BP01 Utilizzo di un processo per la gestione di eventi, incidenti e problemi

La capacità di gestire in modo efficiente eventi, incidenti e problemi è fondamentale per mantenere l'integrità e le prestazioni del carico di lavoro. È essenziale riconoscere e comprendere le differenze tra questi elementi per sviluppare una strategia di risposta e risoluzione efficace. Stabilire e seguire un processo ben definito per ogni aspetto facilita la gestione rapida ed efficace da parte del tuo team di qualsiasi sfida operativa che si presenti.

Risultato desiderato: la tua organizzazione gestisce efficacemente eventi operativi, incidenti e problemi attraverso processi ben documentati e archiviati a livello centrale. Questi processi vengono costantemente aggiornati per riflettere le modifiche, semplificando la gestione e mantenendo l'affidabilità del servizio e delle prestazioni dei carichi di lavoro elevata.

Anti-pattern comuni:

- Rispondi in modo reattivo, anziché proattivo, agli eventi.
- Vengono adottati approcci incoerenti a diversi tipi di eventi o incidenti.
- La tua organizzazione non effettua analisi e non impara dagli incidenti per prevenire eventi futuri.

Vantaggi dell'adozione di questa best practice:

- Processi di risposta semplificati e standardizzati.
- Riduzione dell'impatto degli incidenti su servizi e clienti.
- Risoluzione rapida dei problemi.
- Miglioramento continuo dei processi operativi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

L'implementazione di questa best practice prevede la registrazione degli eventi dei carichi di lavoro. Per la gestione di incidenti e problemi, è necessario ricorrere ai processi. I processi sono documentati, condivisi e aggiornati con frequenza. I problemi vengono identificati, classificati in base alla priorità e corretti.

Informazioni su eventi, incidenti e problemi

- Eventi: un evento è l'adempimento di un'azione, un'occorrenza o un cambiamento di stato. Gli eventi possono essere pianificati o non pianificati e possono avere origine all'interno o all'esterno del carico di lavoro.
- Incidenti: gli incidenti sono eventi che richiedono una risposta, come interruzioni non pianificate o il peggioramento della qualità del servizio. Rappresentano interruzioni che richiedono un'attenzione immediata al fine di ripristinare il normale funzionamento del carico di lavoro.
- Problemi: i problemi sono le cause alla base di uno o più incidenti. Identificare e risolvere i problemi implica approfondire gli incidenti per prevenire eventi futuri.

Passaggi dell'implementazione

Eventi

1. Monitora gli eventi:
   - Implementa l'osservabilità e sfrutta l'osservabilità del carico di lavoro.
   - Le azioni di monitoraggio intraprese da un utente, ruolo o servizio AWS vengono registrate come eventi in AWS CloudTrail.
   - Rispondi alle modifiche operative delle tue applicazioni in tempo reale con Amazon EventBridge.
   - Valuta, monitora e registra continuamente le modifiche alla configurazione delle risorse con AWS Config.
2. Crea processi:
   - Sviluppa un processo per valutare quali eventi sono significativi e richiedono di essere monitorati. Ciò comporta l'impostazione di soglie e parametri per le attività normali e anomale.
   - Determina i criteri in base ai quali un evento viene segnalato come un incidente, ad esempio, la gravità dell'evento, l'impatto sugli utenti o la deviazione dal comportamento previsto.
   - Rivedi regolarmente i processi di monitoraggio e risposta agli eventi. Ciò include l'analisi degli incidenti passati, l'adeguamento delle soglie e il perfezionamento dei meccanismi di avviso.

Incidenti

1. Rispondi agli incidenti:
   - Usa gli approfondimenti degli strumenti di osservabilità per identificare e rispondere rapidamente agli incidenti.
   - Implementa AWS Systems Manager Ops Center per aggregare, organizzare e dare priorità agli elementi operativi e agli incidenti.

- Utilizza servizi come Amazon CloudWatch e AWS X-Ray per analisi e risoluzione dei problemi più approfondite.
- Prendi in considerazione AWS Managed Services (AMS) per una gestione degli incidenti avanzata, sfruttandone funzionalità proattive, preventive e investigative. AMS estende il supporto operativo con servizi come monitoraggio, rilevamento, risposta agli incidenti e gestione della sicurezza.
- Per i clienti del supporto Enterprise, AWS Incident Detection and Response offre un monitoraggio proattivo continuo e la gestione degli incidenti per i carichi di lavoro di produzione.

2. Crea un processo di gestione degli incidenti:
   - Definisci un processo strutturato di gestione degli incidenti, che includa ruoli, protocolli di comunicazione e passaggi per la risoluzione chiari.
   - Integra la gestione degli incidenti con strumenti come Amazon Q Developer nelle applicazioni di chat per garantire l'efficienza nella risposta e nel coordinamento.
   - Suddividi in categorie gli incidenti in base alla gravità, con piani di risposta agli incidenti predefiniti per ciascuna di esse.
3. Apprendi e migliora:
   - Effettua analisi post-incidente per comprendere le cause principali e l'efficacia della risoluzione.
   - Aggiorna e migliora continuamente i piani di risposta in base alle revisioni e alle pratiche in evoluzione.
   - Documenta e condividi le lezioni apprese tra i team per migliorare la resilienza operativa.
   - I clienti del supporto Enterprise possono rivolgersi al proprio Technical Account Manager per il workshop sulla gestione degli incidenti. Questo workshop guidato consente di verificare il piano di risposta agli incidenti esistente e ti aiuta a individuare eventuali aree da migliorare.

Problemi

1. Identifica i problemi:
   - Utilizza i dati degli incidenti passati per identificare modelli ricorrenti che potrebbero indicare la presenza di problemi sistemici più profondi.
   - Sfrutta strumenti come AWS CloudTrail e Amazon CloudWatch per l'analisi delle tendenze e l'individuazione dei problemi alla base.
   - Coinvolgi team interfunzionali, ad esempio i team dediti alle operazioni, allo sviluppo e i reparti aziendali, per ottenere prospettive diverse sulle cause principali.

2. Crea un processo di gestione dei problemi:
    - Sviluppa un processo strutturato per la gestione dei problemi, concentrandoti su soluzioni a lungo termine piuttosto che su correzioni rapide.
    - Incorpora tecniche di analisi delle cause principali (RCA) per indagare e comprendere le cause alla base degli incidenti.
    - Aggiorna policy e procedure operative e l'infrastruttura in base agli esiti per prevenire il ripetersi degli incidenti.
3. Continua a migliorare:
    - Promuovi una cultura di apprendimento e miglioramento continui, incoraggiando i team a identificare e affrontare in modo proattivo i problemi potenziali.
    - Analizza e rivedi regolarmente i processi e gli strumenti di gestione dei problemi per allinearli agli scenari aziendali e tecnologici in evoluzione.
    - Condividi approfondimenti e best practice in tutta l'organizzazione per creare un ambiente operativo più resiliente ed efficiente.
4. Integra Supporto AWS:
    - Consulta le risorse di supporto AWS, come AWS Trusted Advisor, per indicazioni proattive e suggerimenti in merito all'ottimizzazione.
    - I clienti del supporto Enterprise hanno a disposizione programmi dedicati, come AWS Countdown, per ricevere assistenza durante gli eventi critici.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS04-BP01 Identifica gli indicatori chiave di prestazione
- OPS04-BP02 Implementare la telemetria delle applicazioni
- OPS07-BP03 Utilizzo di runbook per eseguire le procedure
- OPS07-BP04 Utilizzo dei playbook per analizzare i problemi
- OPS08-BP01 Analizza le metriche del carico di lavoro
- OPS11-BP02 Eseguire l'analisi post-incidente

Documenti correlati:

- AWS Security Incident Response Guide
- Rilevamento e risposta agli incidenti di AWS
- AWS Cloud Adoption Framework: Operations Perspective - Incident and problem management
- Incident Management in the Age of DevOps and SRE
- PagerDuty - What is Incident Management?

Video correlati:

- Top incident response tips from AWS
- AWS re:Invent 2022 - The Amazon Builders' Library: 25 yrs of Amazon operational excellence
- AWS re:Invent 2022 - AWS Incident Detection and Response (SUP201)
- Introducing Incident Manager from AWS Systems Manager

Esempi correlati:

- AWS Proactive Services: workshop sulla gestione degli incidenti
- How to Automate Incident Response with PagerDuty and Strumento di gestione degli incidenti AWS Systems Manager
- Engage Incident Responders with the On-Call Schedules in Strumento di gestione degli incidenti AWS Systems Manager
- Improve the Visibility and Collaboration during Incident Handling in Strumento di gestione degli incidenti AWS Systems Manager
- Incident reports and service requests in AMS

Servizi correlati:

- Amazon EventBridge

OPS10-BP02 Definizione di un processo per ogni avviso

Stabilire un processo chiaro e definito per ogni avviso nel sistema è essenziale per una gestione degli incidenti efficace ed efficiente. Questa pratica garantisce che ogni avviso porti a una risposta specifica e attuabile, migliorando l'affidabilità e la reattività delle operazioni.

Risultato desiderato: ogni avviso avvia un piano di risposta specifico e ben definito. Ove possibile, le risposte sono automatizzate e dotate di una chiara titolarità e di un percorso di escalation definito. Gli avvisi sono collegati a una base di conoscenze aggiornata, in modo che qualsiasi operatore sia in grado di rispondere in modo coerente ed efficace. Le risposte sono rapide e uniformi su tutta la linea, migliorando l'efficienza e l'affidabilità operativa.

Anti-pattern comuni:

- Gli avvisi non hanno un processo di risposta predefinito, il che porta a risoluzioni improvvisate e tardive.
- Il sovraccarico di avvisi comporta che gli avvisi importanti vengano trascurati.
- Gli avvisi vengono gestiti in modo incoerente a causa della mancanza di titolarità e responsabilità chiare.

Vantaggi dell'adozione di questa best practice:

- Creazione solo di avvisi utilizzabili, con conseguente riduzione dell'affaticamento da avvisi.
- Riduzione del tempo medio di risoluzione (MTTR) per problemi operativi.
- Riduzione del tempo medio di indagine (MTTI), il che aiuta a ridurre l'MTTR.
- Migliore capacità di scalare le risposte operative.
- Maggiore coerenza e affidabilità nella gestione degli eventi operativi.

Ad esempio, disponi di un processo definito per gli eventi di AWS Health per gli account critici, compresi gli allarmi delle applicazioni, i problemi operativi e gli eventi del ciclo di vita pianificati (come l'aggiornamento delle versioni di Amazon EKS prima dell'aggiornamento automatico dei cluster) e fornisci ai team la possibilità di monitorare attivamente, comunicare e rispondere a questi eventi. Queste azioni aiutano a prevenire le interruzioni del servizio causate da modifiche lato AWS o a mitigarle più rapidamente quando si verificano problemi imprevisti.

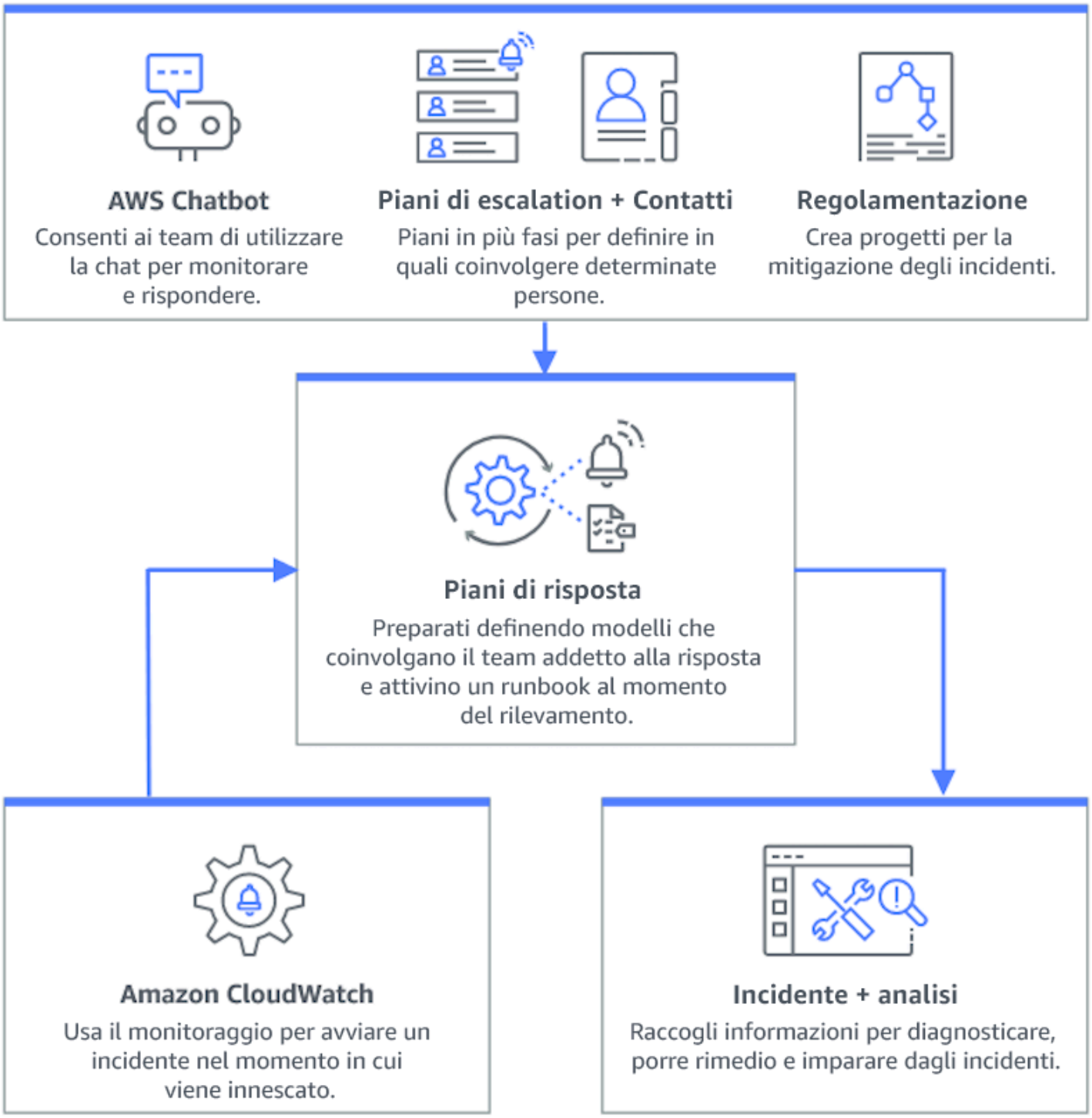
Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Avere un processo per ogni avviso implica stabilire un piano di risposta chiaro per ciascun avviso, automatizzare le risposte ove possibile e perfezionare continuamente questi processi in base al feedback operativo e all'evoluzione dei requisiti.

Passaggi dell'implementazione

Il diagramma seguente illustra il flusso di lavoro di gestione degli incidenti all'interno di Strumento di gestione degli incidenti AWS Systems Manager. È progettato per rispondere rapidamente ai problemi operativi creando automaticamente incidenti in risposta a eventi specifici che si verificano in Amazon CloudWatch o Amazon EventBridge. Quando viene creato automaticamente o manualmente un incidente, Incident Manager centralizza la gestione dell'incidente, organizza le informazioni pertinenti sulle risorse AWS e avvia piani di risposta predefiniti. Ciò include l'esecuzione dei runbook di automazione di Systems Manager per un'azione immediata e la creazione di un elemento di lavoro operativo principale in OpsCenter per tenere traccia delle attività e delle analisi correlate. Questo processo semplificato accelera e coordina la risposta agli incidenti in tutto l'ambiente AWS.



1. Utilizza allarmi compositi: crea allarmi compositi in CloudWatch per raggruppare allarmi correlati, così da ridurre il rumore e consentire risposte più significative.
2. Resta aggiornato con AWS Health: AWS Health è la fonte autorevole di informazioni sull'integrità delle risorse Cloud AWS. Utilizza AWS Health per visualizzare e ricevere notifiche su eventuali

eventi di servizio in corso e modifiche imminenti, come gli eventi pianificati del ciclo di vita, in modo da poter adottare misure per mitigare gli impatti.

a. Crea notifiche di eventi AWS Health personalizzati per i canali e-mail e chat con Notifiche all'utente AWS e integra a livello di codice con gli strumenti di monitoraggio e avviso di Amazon EventBridge o l'AWS Health API.
b. Pianifica e monitora i progressi relativi agli eventi sull'integrità che richiedono un'azione integrando con strumenti di gestione delle modifiche o ITSM (come Jira ServiceNow) che potresti già utilizzare tramite Amazon EventBridge o l'API AWS Health.
c. Se utilizzi AWS Organizations, abilita la visualizzazione dell'organizzazione per AWS Health per aggregare gli eventi AWS Health tra gli account.

3. Integra gli allarmi di Amazon CloudWatch con lo strumento di gestione degli incidenti: configura gli allarmi di CloudWatch per la creazione automatica di incidenti in Strumento di gestione degli incidenti AWS Systems Manager.
4. Integra Amazon EventBridge con Incident Manager: crea regole EventBridge in modo da reagire agli eventi e creare incidenti mediante piani di risposta definiti.
5. Preparati per gli incidenti in Incident Manager:
   - Crea piani di risposta dettagliati in Incident Manager per ciascun tipo di avviso.
   - Stabilisci canali di chat tramite Amazon Q Developer nelle applicazioni di chat collegato ai piani di risposta nello strumento di gestione degli incidenti, semplificando la comunicazione in tempo reale durante gli incidenti su piattaforme come Slack, Microsoft Teams e Amazon Chime.
   - Integra i runbook di Systems Manager Automation in Incident Manager per fornire risposte automatiche agli incidenti.

Risorse

Best practice correlate:

- OPS04-BP01 Identifica gli indicatori chiave di prestazione
- OPS08-BP04 Creare avvisi fruibili

Documenti correlati:

- AWS Cloud Adoption Framework: Operations Perspective - Incident and problem management
- Using Amazon CloudWatch alarms
- Setting up Strumento di gestione degli incidenti AWS Systems Manager

- Preparing for incidents in Incident Manager

Video correlati:

- Top incident response tips from AWS
- re:Invent 2023 | Manage resource lifecycle events at scale with AWS Health

Esempi correlati:

- Workshop AWS, Strumento di gestione degli incidenti AWS Systems Manager: Automate incident response to security events

OPS10-BP03 Definizione della priorità degli eventi operativi in base agli effetti sul business

Rispondere tempestivamente agli eventi operativi è fondamentale, ma non tutti gli eventi sono uguali. Quando si assegnano le priorità in base all'impatto sul business, si dà la priorità anche alla risoluzione di eventi che possono avere conseguenze significative, come la compromissione della sicurezza, perdite finanziarie, violazioni normative o danni alla reputazione.

Risultato desiderato: la priorità delle risposte agli eventi operativi si basa sul potenziale impatto dell'evento su operazioni e obiettivi di business. Ciò rende le risposte efficienti ed efficaci.

Anti-pattern comuni:

- Ogni evento viene trattato con lo stesso livello di urgenza, generando confusione e ritardi nell'affrontare le criticità.
- Non è possibile distinguere tra eventi ad alto e basso impatto, con conseguente errata allocazione delle risorse.
- L'organizzazione non dispone di un chiaro framework di assegnazione delle priorità, il che genera risposte incoerenti agli eventi operativi.
- Agli eventi viene assegnata la priorità in base all'ordine in cui vengono segnalati piuttosto che al loro impatto sui risultati aziendali.

Vantaggi dell'adozione di questa best practice:

- Assicura che la risposta si concentri in primo luogo sulle funzioni aziendali critiche, riducendo al minimo i danni potenziali.

- Migliora l'allocazione delle risorse durante più eventi simultanei.
- Migliora la capacità dell'organizzazione di mantenere la fiducia e soddisfare i requisiti normativi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Di fronte a molteplici eventi operativi, è essenziale un approccio strutturato alla definizione delle priorità basato sull'impatto e sull'urgenza. Questo approccio consente di prendere decisioni informate, indirizzare gli sforzi laddove sono più necessari e mitigare il rischio per la continuità aziendale.

Passaggi dell'implementazione

1. Valuta l'impatto: sviluppa un sistema di classificazione per valutare la gravità degli eventi in termini di potenziale impatto sulle operazioni e sugli obiettivi di business. L'esempio seguente mostra le categorie di impatto:

Livello di impatto	Descrizione
Elevata	Coinvolge molti dipendenti o clienti, ha un elevato impatto finanziario, genera un elevato danno alla reputazione o lesioni.
Media	Coinvolge un gruppo di dipendenti o clienti, ha un impatto finanziario moderato o genera un danno alla reputazione moderato.
Bassa	Coinvolge singoli dipendenti o clienti, ha un basso impatto finanziario o genera un danno alla reputazione di lieve entità.

2. Valuta l'urgenza: definisci i livelli di urgenza in base alla rapidità con cui un evento deve ricevere una risposta, considerando fattori come la sicurezza, le implicazioni finanziarie e accordi sul livello di servizio (SLA). L'esempio seguente illustra le categorie di urgenza:

Livello di urgenza	Descrizione
Elevata	Produce danni che aumentano in maniera esponenziale, incide su un lavoro sensibile al

Livello di urgenza	Descrizione
	fattore tempo, escalation imminente, interessa utenti o gruppi VIP.
Media	Produce danni che aumentano nel tempo oppure interessa un singolo utente o gruppo VIP.
Bassa	Produce danni marginali che aumentano nel tempo o incide su lavori non sensibili al fattore tempo.

3. Crea una matrice di prioritizzazione:
    - Usa una matrice per incrociare impatto e urgenza, assegnando livelli di priorità a diverse combinazioni.
    - Rendi la matrice accessibile e comprensibile da tutti i membri del team responsabili delle risposte agli eventi operativi.
    - La seguente matrice di esempio mostra la gravità dell'incidente in base all'urgenza e all'impatto:

Urgenza e impatto	Elevata	Media	Bassa
Elevata	Critica	Urgente	Elevata
Media	Urgente	Elevata	Normale
Bassa	Elevata	Normale	Bassa

4. Predisponi formazione e comunicazione: forma i team di risposta sulla matrice di prioritizzazione e sull'importanza di attenersi a essa durante un evento. Comunica il processo di definizione delle priorità a tutte le parti interessate per stabilire aspettative chiare.
5. Integra con la risposta agli incidenti:
    - Incorpora la matrice di prioritizzazione nei tuoi piani e strumenti di risposta agli incidenti.
    - Automatizza la classificazione e la prioritizzazione degli eventi, ove possibile, per accelerare i tempi di risposta.

- I clienti del supporto Enterprise, possono sfruttare AWS Incident Detection and Response che garantisce il monitoraggio proattivo 24 ore su 24, 7 giorni su 7, oltre alla gestione degli incidenti per i carichi di lavoro di produzione.

6. Rivedi e adatta: rivedi regolarmente l'efficacia del processo di definizione delle priorità e apporta modifiche in base al feedback e ai cambiamenti nell'ambiente aziendale.

Risorse

Best practice correlate:

- OPS03-BP03 L'escalation è incoraggiata
- OPS08-BP04 Creare avvisi fruibili
- OPS09-BP01 Misura gli obiettivi operativi e i KPI con le metriche

Documenti correlati:

- Atlassian - Understanding incident severity levels
- IT Process Map - Checklist Incident Priority

OPS10-BP04 Definizione dei percorsi di escalation

Stabilisci percorsi di escalation chiari all'interno dei tuoi protocolli di risposta agli incidenti per facilitare un'azione tempestiva ed efficace. Ciò include la specificazione delle richieste relative all'escalation, la descrizione dettagliata del processo di escalation e la preapprovazione delle azioni per accelerare il processo decisionale e ridurre il tempo medio di risoluzione (MTTR).

Risultato desiderato: un processo strutturato ed efficiente che inoltra gli incidenti al personale appropriato, riducendo al minimo i tempi di risposta e l'impatto.

Anti-pattern comuni:

- La mancanza di chiarezza in merito alle procedure di ripristino genera risposte improvvisate in caso di incidenti critici.
- L'assenza di autorizzazioni e titolarità definite comporta ritardi quando è necessaria un'azione urgente.
- Le parti interessate e i clienti non sono informati nei tempi attesi.
- Le decisioni importanti subiscono ritardi.

Vantaggi dell'adozione di questa best practice:

- Risposta semplificata agli incidenti tramite procedure di escalation predefinite.
- Tempi di inattività ridotti con azioni preapprovate e titolarità chiara.
- Migliore allocazione delle risorse e adeguamenti del livello di supporto in base alla gravità degli incidenti.
- Migliore comunicazione con le parti interessate e i clienti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I percorsi di escalation correttamente definiti sono fondamentali per una risposta rapida agli incidenti. Strumento di gestione degli incidenti AWS Systems Manager supporta l'impostazione di piani di escalation strutturati e di pianificazioni della reperibilità, che avvisano il personale pertinente preparandolo ad agire in caso di incidenti.

Passaggi dell'implementazione

1. Configura le richieste di escalation: imposta allarmi CloudWatch per la creazione di un incidente in Strumento di gestione degli incidenti AWS Systems Manager.
2. Imposta la pianificazione della reperibilità: crea la pianificazione della reperibilità in Incident Manager, in linea con i tuoi percorsi di escalation. Fornisci al personale di turno le autorizzazioni e gli strumenti necessari per agire rapidamente.
3. Procedure di escalation dettagliate:
   - Determina le condizioni specifiche in base alle quali un incidente deve essere inoltrato.
   - Crea piani di escalation in Incident Manager.
   - I canali di escalation devono consistere in un contatto o in una pianificazione della reperibilità.
   - Definisci i ruoli e le responsabilità del team a ogni livello di escalation.
4. Approva preventivamente le azioni di mitigazione: collabora con i responsabili delle decisioni per approvare preventivamente le azioni per gli scenari previsti. Sfrutta i runbook di Systems Manager Automation integrati con Incident Manager per velocizzare la risoluzione degli incidenti.
5. Specifica la proprietà: identifica chiaramente i proprietari interni per ogni fase del percorso di escalation.
6. Fornisci dettagli in merito alle escalation a terze parti:
   - Documenta gli accordi sul livello di servizio (SLA) di terze parti e adeguali agli obiettivi interni.

- Stabilisci protocolli chiari per la comunicazione con i fornitori durante gli incidenti.
- Integra i contatti dei fornitori negli strumenti di gestione degli incidenti per l'accesso diretto.
- Conduci regolarmente esercitazioni che includano scenari di risposta di terze parti.
- Mantieni le informazioni sulle escalation dei fornitori ben documentate e facilmente accessibili.

7. Esegui formazione e test per i piani di escalation: forma il tuo team sul processo di escalation e conduci regolarmente esercitazioni di risposta agli incidenti o giornate di gioco. I clienti del supporto Enterprise possono richiedere un workshop sulla gestione degli incidenti.
8. Continua a migliorare: verifica regolarmente l'efficacia dei tuoi percorsi di escalation. Aggiorna i tuoi processi in base alle lezioni apprese dalle analisi degli incidenti e dal feedback continuo.

Livello di impegno per il piano di implementazione: moderato

Risorse

Best practice correlate:

- OPS08-BP04 Creare avvisi fruibili
- OPS10-BP02 Definizione di un processo per ogni avviso
- OPS11-BP02 Eseguire l'analisi post-incidente

Documenti correlati:

- Piani di escalation di Strumento di gestione degli incidenti AWS Systems Manager
- Working with on-call schedules in Incident Manager
- Creating and Managing Runbooks
- Temporary elevated access management with AWS IAM Identity Center
- Atlassian - Escalation policies for effective incident management

OPS10-BP05 Definizione di un piano di comunicazione con i clienti per eventi che incidono sul servizio

Una comunicazione efficace durante gli eventi che incidono sul servizio è fondamentale per mantenere la fiducia e la trasparenza con i clienti. Un piano di comunicazione ben definito sostiene la comunicazione rapida e chiara di informazioni all'interno e all'esterno dell'organizzazione durante gli incidenti.

Risultato desiderato:

- Un solido piano di comunicazione che informa efficacemente i clienti e le parti interessate durante gli eventi che influiscono sul servizio.
- Trasparenza nella comunicazione per creare fiducia e ridurre la preoccupazione dei clienti.
- Riduzione al minimo dell'impatto che gli eventi che incidono sul servizio hanno sull'esperienza del cliente e sulle operazioni aziendali.

Anti-pattern comuni:

- Una comunicazione inadeguata o in ritardo genera confusione e insoddisfazione nei clienti.
- Una messaggistica eccessivamente tecnica o vaga impedisce la comunicazione dell'impatto effettivo sugli utenti.
- È assente una strategia di comunicazione predefinita, con conseguente messaggistica incoerente e reattiva.

Vantaggi dell'adozione di questa best practice:

- Maggiore fiducia e soddisfazione dei clienti attraverso una comunicazione chiara e proattiva.
- Riduzione del carico operativo per i team di supporto grazie alla risoluzione preventiva delle preoccupazioni dei clienti.
- Maggiore efficienza di gestione e risoluzione degli incidenti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

La creazione di un piano di comunicazione completo per gli eventi che incidono sul servizio implica prendere in considerazione molteplici aspetti, dalla scelta dei canali giusti alla creazione del messaggio e del tono. Il piano deve essere adattabile, scalabile e soddisfare diversi scenari di interruzione del servizio.

Passaggi dell'implementazione

1. Definisci ruoli e responsabilità:
    - Assegna a un responsabile degli incidenti gravi la supervisione delle attività di risposta agli incidenti.

- Designa un responsabile delle comunicazioni dedicato al coordinamento di tutte le comunicazioni esterne e interne.
- Includi il responsabile dell'assistenza per fornire una comunicazione coerente attraverso ticket di supporto.

2. Identifica i canali di comunicazione: seleziona canali come chat aziendale, e-mail, SMS, social media, notifiche in-app e pagine di stato. Questi canali devono essere resilienti e in grado di operare in maniera indipendente durante gli eventi che incidono sul servizio.
3. Comunica in modo rapido, chiaro e regolare con i clienti:
   - Sviluppa modelli per vari scenari di compromissione del servizio, focalizzandoti sulla semplicità e sui dettagli essenziali. Includi informazioni sul problema relativo al servizio, sui tempi di risoluzione previsti e sull'impatto.
   - Usa Amazon Pinpoint per avvisare i clienti tramite notifiche push, notifiche in-app, e-mail, SMS, messaggi vocali e messaggi su canali personalizzati.
   - Usa Amazon Simple Notification Service (Amazon SNS) per avvisare gli abbonati in modo programmatico o tramite e-mail, notifiche push su dispositivi mobili e SMS.
   - Comunica lo stato tramite pannelli di controllo, condividendone pubblicamente uno di Amazon CloudWatch.
   - Incoraggia il coinvolgimento sui social media:
     - Monitora attivamente i social media per comprendere il sentimento dei clienti.
     - Pubblica post su piattaforme di social media per aggiornare il pubblico e coinvolgere la comunità.
     - Prepara modelli per una comunicazione coerente e chiara sui social media.
4. Coordina la comunicazione interna: implementa protocolli interni utilizzando strumenti come Amazon Q Developer per migliorare il coordinamento e la comunicazione tra i team. Usa i pannelli di controlli di CloudWatch per comunicare lo stato.
5. Orchestra la comunicazione con strumenti e servizi dedicati:
   - Usa Strumento di gestione degli incidenti AWS Systems Manager con Amazon Q Developer per configurare canali di chat dedicati per la comunicazione interna e il coordinamento in tempo reale durante gli incidenti.
   - Usa i runbook Strumento di gestione degli incidenti AWS Systems Manager per automatizzare le notifiche ai clienti durante gli incidenti tramite Amazon Pinpoint, Amazon SNS o strumenti di terze parti come le piattaforme di social media.

- Incorpora i flussi di lavoro di approvazione all'interno dei runbook per rivedere e autorizzare tutte le comunicazioni esterne prima dell'invio.

6. Fai pratica e migliora:
   - Tieni corsi di formazione sull'uso di strumenti e strategie di comunicazione. Responsabilizza i team affinché siano in grado di prendere decisioni tempestive durante gli incidenti.
   - Testa il piano di comunicazione con esercitazioni regolari o giornate di gioco. Usa questi test per perfezionare la messaggistica e valutare l'efficacia dei canali.
   - Implementa meccanismi di feedback per valutare l'efficacia della comunicazione durante gli incidenti. Sviluppa continuamente il piano di comunicazione in base al feedback e alle esigenze mutevoli.

Livello di impegno per il piano di implementazione: elevato

Risorse

Best practice correlate:

- OPS07-BP03 Utilizzo di runbook per eseguire le procedure
- OPS10-BP06 Comunicazione dello stato tramite pannelli di controllo
- OPS11-BP02 Eseguire l'analisi post-incidente

Documenti correlati:

- Atlassian - Incident communication best practices
- Atlassian - How to write a good status update
- PagerDuty - A Guide to Incident Communications

Video correlati:

- Atlassian - Create your own incident communication plan: Incident templates

Esempi correlati:

- Dashboard di AWS Health

OPS10-BP06 Comunicazione dello stato tramite pannelli di controllo

Usa i pannelli di controllo come strumento strategico per trasmettere lo stato operativo e le metriche fondamentali in tempo reale a diversi tipi di pubblico, inclusi team tecnici interni, leader e clienti. Questi pannelli di controllo offrono una rappresentazione visiva centralizzata dello stato del sistema e delle prestazioni aziendali, il che migliora la trasparenza e l'efficienza decisionale.

Risultato desiderato:

- I pannelli di controllo forniscono una visione completa del sistema e delle metriche aziendali rilevanti per le varie parti interessate.
- Le parti interessate possono accedere in modo proattivo alle informazioni operative, il che riduce la necessità di richieste di stato frequenti.
- Migliore processo decisionale in tempo reale durante le normali operazioni e gli incidenti.

Anti-pattern comuni:

- I tecnici che partecipano a una chiamata di gestione degli incidenti hanno bisogno di ricevere aggiornamenti di stato per poter agire rapidamente.
- Affidarsi ai report manuali per la gestione comporta ritardi e potenziali imprecisioni.
- I team operativi vengono spesso interrotti per aggiornamenti sullo stato durante gli incidenti.

Vantaggi dell'adozione di questa best practice:

- Consente alle parti interessate di accedere immediatamente alle informazioni critiche, promuovendo un processo decisionale informato.
- Riduce le inefficienze operative riducendo al minimo i report manuali e le richieste di stato frequenti.
- Aumenta la trasparenza e la fiducia attraverso la visibilità in tempo reale delle prestazioni del sistema e delle metriche aziendali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I pannelli di controllo comunicano efficacemente lo stato dei sistemi e le metriche aziendali e possono essere personalizzati in base alle esigenze di diversi gruppi di destinatari. Strumenti come i pannelli

di controllo di Amazon CloudWatch e Amazon Quick aiutano a creare pannelli di controllo interattivi e in tempo reale per il monitoraggio del sistema e la business intelligence.

Passaggi dell'implementazione

1. Identifica le esigenze delle parti interessate: determina le esigenze in termini di informazioni specifiche dei diversi gruppi di destinatari, come team tecnici, leader e clienti.
2. Scegli gli strumenti giusti: seleziona gli strumenti appropriati come i pannelli di controllo di Amazon CloudWatch per il monitoraggio del sistema e Amazon Quick per la business intelligence interattiva. AWS Health offre un'esperienza pronta all'uso in Dashboard AWS Health, oppure puoi utilizzare gli eventi di stato in Amazon EventBridge o tramite l'API AWS Health per aumentare i pannelli di controllo.
3. Progetta pannelli di controllo efficaci:
    - Progetta pannelli di controllo per presentare in modo chiaro metriche e KPI pertinenti, assicurandoti che siano comprensibili e utilizzabili.
    - Incorpora visualizzazioni a livello di sistema e a livello aziendale, se necessario.
    - Includi pannelli di controllo di alto livello (per ampie panoramiche) e di basso livello (per analisi dettagliate).
    - Integra allarmi automatici all'interno dei pannelli di controllo per evidenziare i problemi critici.
    - Annota i pannelli di controllo con soglie e obiettivi delle metriche importanti per una visibilità immediata.
4. Integra l'origine dati:
    - Utilizza Amazon CloudWatch per aggregare e visualizzare i parametri di vari servizi AWS e i parametri delle query provenienti da altre origini dati, creando in questo modo una visualizzazione unificata dello stato e dei parametri aziendali del tuo sistema.
    - Utilizza funzionalità come Approfondimenti di CloudWatch Logs per le query e la visualizzazione di dati di log provenienti da vari applicazioni e servizi.
    - Usa gli eventi AWS Health per rimanere informato sullo stato operativo e sui problemi operativi confermati dei servizi AWS tramite l'API AWS Health o gli eventi AWS Health su Amazon EventBridge.
5. Fornisci l'accesso self-service:
    - Condividi i pannelli di controllo CloudWatch con le parti interessate pertinenti per l'accesso self-service alle informazioni utilizzando la funzionalità di condivisione dei pannelli di controllo.

- Assicurati che i pannelli di controllo siano facilmente accessibili e contengano informazioni aggiornate in tempo reale.

6. Aggiorna e perfeziona regolarmente:
   - Aggiorna e perfeziona continuamente i panelli di controllo per allinearli alle esigenze aziendali in evoluzione e ai feedback delle parti interessate.
   - Rivedi regolarmente i panelli di controllo per assicurarti che siano sempre pertinenti ed efficaci nella trasmissione delle informazioni necessarie.

Risorse

Best practice correlate:

- OPS08-BP05 Creare dashboard

Documenti correlati:

- Creazione di pannelli di controllo per visibilità operativa
- Using Amazon CloudWatch dashboards
- Create flexible dashboards with dashboard variables
- Sharing CloudWatch dashboards
- Query metrics from other data sources
- Add a custom widget to a CloudWatch dashboard

Esempi correlati:

- One Observability Workshop - Dashboards

OPS10-BP07 Automatizza le risposte agli eventi

L'automazione delle risposte agli eventi è fondamentale per una gestione operativa rapida, coerente e priva di errori. Crea processi semplificati e utilizza strumenti per gestire e rispondere automaticamente agli eventi, riducendo al minimo gli interventi manuali e migliorando l'efficacia operativa.

Risultato desiderato:

- Riduzione degli errori umani e tempi di risoluzione più rapidi grazie all'automazione.
- Gestione degli eventi operativi coerente e affidabile.
- Maggiore efficienza operativa e affidabilità del sistema.

Anti-pattern comuni:

- La gestione manuale degli eventi comporta ritardi ed errori.
- L'automazione viene trascurata nelle attività ripetitive e critiche.
- Le attività manuali ripetitive causano affaticamento da avvisi e la mancata identificazione di problemi critici.

Vantaggi dell'adozione di questa best practice:

- Risposte agli eventi accelerate, riduzione dei tempi di inattività del sistema.
- Operazioni affidabili con gestione automatizzata e coerente degli eventi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Incorpora l'automazione per creare flussi di lavoro operativi efficienti e ridurre al minimo gli interventi manuali.

Passaggi dell'implementazione

1. Identifica le opportunità di automazione: definisci le attività ripetitive da automatizzare, come la risoluzione dei problemi, l'arricchimento dei ticket, la gestione della capacità, la scalabilità, le implementazioni e i test.
2. Identifica i prompt di automazione:
   - Valuta e definisci condizioni o metriche specifiche che avviano risposte automatiche utilizzando le azioni di CloudWatch allarme di Amazon.
   - Usa Amazon EventBridge per rispondere agli eventi nei AWS servizi, nei carichi di lavoro personalizzati e nelle applicazioni SaaS.
   - Prendi in considerazione eventi di avvio come voci di registro specifiche, soglie di metriche prestazionali o cambiamenti di stato nelle risorse. AWS
3. Implementa l'automazione basata sugli eventi:

- Utilizza i runbook di AWS Systems Manager automazione per semplificare le attività di manutenzione, implementazione e correzione.
- La creazione di incidenti in Incident Manager raccoglie e aggiunge automaticamente dettagli sulle AWS risorse coinvolte nell'incidente.
- Monitora in modo proattivo le quote utilizzando Quota Monitor for AWS.
- Regola in automatico la capacità di AWS Auto Scaling così da mantenere disponibilità e prestazioni.
- Automatizza le pipeline di sviluppo con Amazon. CodeCatalyst
- Smoke testa o monitora continuamente gli endpoint utilizzando il monitoraggio sintetico. APIs

4. Esegui la mitigazione del rischio attraverso l'automazione:
   - Implementa le risposte di sicurezza automatizzate per affrontare in modo rapido i rischi.
   - Utilizza AWS Systems Manager State Manager per ridurre la deviazione delle configurazioni.
   - Risolvi le risorse non conformi con. Regole di AWS Config

Livello di impegno per il piano di implementazione: elevato

Risorse

Best practice correlate:

- OPS08-BP04 Creare avvisi fruibili
- OPS10-BP02 Definizione di un processo per ogni avviso

Documenti correlati:

- Using Systems Manager Automation runbooks with Incident Manager
- Creating incidents in Incident Manager
- AWS quote di servizio
- Monitor resource usage and send notifications when approaching quotas
- AWS Auto Scaling
- Che cos'è Amazon CodeCatalyst?
- Utilizzo degli CloudWatch allarmi Amazon
- Utilizzo delle azioni di CloudWatch allarme di Amazon

- Correzione delle risorse non conformi con Regole di AWS Config
- Creating metrics from log events using filters
- AWS Systems Manager State Manager

Video correlati:

- Crea runbook di automazione con AWS Systems Manager
- Come automatizzare le operazioni IT su AWS
- AWS Security Hub CSPM regole di automazione
- Avvia rapidamente il tuo progetto software con CodeCatalyst i blueprints di Amazon

Esempi correlati:

- CodeCatalyst Tutorial Amazon: creazione di un progetto con il modello di applicazione Web moderno a tre livelli
- One Observability Workshop
- Respond to incidents using Incident Manager

Evoluzione

Domanda

- OPS 11. In che modo fai evolvere le operazioni?

OPS 11. In che modo fai evolvere le operazioni?

Dedica tempo e risorse per ottenere un miglioramento incrementale pressoché continuo, per far evolvere l'efficacia e l'efficienza delle tue operazioni.

Best practice

- OPS11-BP01 Definizione di un processo per il miglioramento continuo
- OPS11-BP02 Eseguire l'analisi post-incidente
- OPS11-BP03 Implementazione di circuiti di feedback
- OPS11-BP04 Eseguire la gestione della conoscenza

- OPS11-BP05 Definizione dei fattori che promuovono il miglioramento
- OPS11-BP06 Validare gli approfondimenti
- OPS11-BP07 Revisioni delle metriche di Perform operations
- OPS11-BP08 Documentazione e condivisione delle conoscenze acquisite
- OPS11-BP09 Dedica tempo per apportare miglioramenti

OPS11-BP01 Definizione di un processo per il miglioramento continuo

Valuta il carico di lavoro rispetto alle best practice dell'architettura interna ed esterna. Effettua revisioni frequenti e deliberate del carico di lavoro. Dai priorità alle opportunità di miglioramento nella cadenza di sviluppo del software.

Risultato desiderato:

- Analizza di frequente il carico di lavoro rispetto alle best practice dell'architettura.
- Stabilisci per le opportunità di miglioramento la stessa priorità che assegni alle funzionalità del processo di sviluppo software.

Anti-pattern comuni:

- Non hai condotto una revisione dell'architettura del carico di lavoro da quando è stato implementato diversi anni fa.
- Assegni una priorità inferiore alle opportunità di miglioramento. Rispetto alle nuove funzionalità, queste opportunità rimangono nel backlog.
- Non esiste uno standard per l'implementazione delle modifiche alle best practice per l'organizzazione.

Vantaggi dell'adozione di questa best practice:

- Il carico di lavoro è aggiornato sulla base delle best practice di architettura.
- Fai evolvere il carico di lavoro in modo intenzionale.
- Puoi utilizzare le best practice dell'organizzazione per migliorare tutti i carichi di lavoro.
- Ottieni guadagni marginali che hanno un impatto cumulativo, con un incremento dell'efficienza.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Effettui di frequente la revisione dell'architettura del carico di lavoro. Utilizzi le best practice interne ed esterne per valutare il carico di lavoro e identificare le opportunità di miglioramento. Dai priorità alle opportunità di miglioramento nella cadenza di sviluppo del software.

Passaggi dell'implementazione

1. Esegui la revisione periodica dell'architettura del carico di lavoro di produzione secondo una frequenza concordata. Utilizza uno standard architettonico documentato che includa best practice specifiche di AWS.
   a. Usa gli standard definiti internamente per queste revisioni. Se non disponi di standard interni, usa il Framework AWS Well-Architected.
   b. Utilizza AWS Well-Architected Tool per creare un obiettivo personalizzato delle best practice interne e condurre la revisione dell'architettura.
   c. Contatta un AWS Solution Architect o Technical Account Manager per condurre una revisione guidata di Framework Well-Architected del carico di lavoro.
2. Dai priorità alle opportunità di miglioramento identificate durante la revisione nel processo di sviluppo del software.

Livello di impegno per il piano di implementazione: basso Puoi usare il Framework AWS Well-Architected per eseguire la revisione annuale dell'architettura.

Risorse

Best practice correlate:

- OPS11-BP02 Esecuzione di analisi post-incidente
- OPS11-BP08 Documentazione e condivisione delle conoscenze acquisite
- OPS04 Come si implementa l'osservabilità nel carico di lavoro?

Documenti correlati:

- AWS Well-Architected Tool - Custom lenses
- Whitepaper AWS Well-Architected: il processo di revisione
- Customize Well-Architected Reviews using Custom Lenses and the AWS Well-Architected Tool

- Implementing the AWS Well-Architected Custom Lens lifecycle in your organization

Video correlati:

- AWS re:Invent 2023 - Scaling AWS Well-Architected best practices across your organization

Esempi correlati:

- AWS Well-Architected Tool

OPS11-BP02 Eseguire l'analisi post-incidente

Esamina gli eventi che influiscono sui clienti e identifica i fattori che contribuiscono e le azioni preventive. Utilizza queste informazioni per sviluppare modi per limitare o prevenire il ripetersi degli incidenti. Sviluppa procedure per attivare risposte rapide ed efficaci. Comunica i fattori che hanno contribuito al presentarsi dell'imprevisto e le azioni correttive secondo necessità, specificamente mirate per il pubblico di destinazione.

Risultato desiderato:

- Stabilisci processi di gestione degli incidenti che includono l'analisi post-incidente.
- Hai a disposizione piani di osservabilità per raccogliere dati sugli eventi.
- Con questi dati comprendi e raccogli metriche che supportano il tuo processo di analisi post-incidente.
- Impari dagli incidenti per migliorare i risultati futuri.

Anti-pattern comuni:

- Sei amministratore di un server di applicazioni. Circa ogni 23 ore e 55 minuti tutte le sessioni attive vengono terminate. Hai tentato di identificare ciò che non va a buon fine sul server di applicazioni. Sospetti che potrebbe trattarsi di un problema di rete, ma non riesci a ottenere la collaborazione dal team di rete perché i suoi membri sono troppo occupati per supportarti. Ti manca un processo predefinito da seguire per ottenere supporto e raccogliere le informazioni necessarie per stabilire che cosa sta accadendo.
- Si è verificata una perdita di dati all'interno del carico di lavoro. Questa è la prima volta che si è verificata e la causa non è immediatamente identificabile. Decidi che non è importante perché puoi

ricreare i dati. La perdita di dati inizia a verificarsi con maggiore frequenza e influisce sui clienti. Questo comporta inoltre un ulteriore onere operativo quando ripristini i dati mancanti.

Vantaggi dell'adozione di questa best practice:

- Disponendo di un processo predefinito per determinare i componenti, le condizioni, le azioni e gli eventi che hanno contribuito a un incidente, sei in grado di identificare le opportunità di miglioramento.
- Utilizzi i dati dell'analisi post-incidente per apportare miglioramenti.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Utilizza un processo per determinare i fattori determinanti. Esamina tutti gli incidenti che influiscono sul cliente. Predisponi un processo per identificare e documentare i fattori che contribuiscono a un incidente, in modo da sviluppare azioni di mitigazione in grado di limitare o impedire il suo ripetersi e per sviluppare procedure che consentano risposte rapide ed efficaci. Comunica le cause principali degli incidenti in modo appropriato e personalizza la comunicazione in base al pubblico di destinazione. Condividi quanto appreso in maniera aperta all'interno della tua organizzazione.

Passaggi dell'implementazione

1. Raccogli metriche come le modifiche all'implementazione e alla configurazione, l'ora di inizio dell'incidente, l'ora dell'allarme, dell'intervento, dell'inizio della mitigazione e il tempo di risoluzione dell'incidente.
2. Descrivi i momenti fondamentali sulla linea temporale per comprendere gli eventi dell'incidente.
3. Poniti le seguenti domande:
    a. Potresti migliorare il tempo di rilevamento?
    b. Sono presenti aggiornamenti alle metriche e agli allarmi che permettono di rilevare l'incidente prima?
    c. Puoi migliorare i tempi di diagnosi?
    d. Sono presenti aggiornamenti ai tuoi piani di risposta o di escalation che potrebbero coinvolgere prima i team di risposta corretti?
    e. Puoi migliorare il tempo necessario per la mitigazione?
    f. Ci sono passaggi del runbook o del playbook che potresti aggiungere o migliorare?

g. È possibile prevenire che si verifichino incidenti futuri?

4. Crea liste di controllo e azioni. Monitora ed esegui tutte le azioni.

Livello di impegno per il piano di implementazione: medio

Risorse

Best practice correlate:

- OPS11-BP01 Definizione di un processo per il miglioramento continuo
- OPS4 - Implementare l'osservabilità

Documenti correlati:

- Performing a post-incident analysis in Incident Manager
- Revisione della prontezza operativa

OPS11-BP03 Implementazione di circuiti di feedback

I cicli di feedback forniscono informazioni fruibili che guidano il processo decisionale. Vanno creati nelle procedure e nei carichi di lavoro per identificare i problemi e le aree che necessitano di miglioramenti. Inoltre, convalidano gli investimenti effettuati nei miglioramenti. Questi cicli di feedback sono la base per migliorare continuamente il carico di lavoro.

Sono due le categorie dei cicli di feedback: feedback immediato e analisi retrospettiva. Il feedback immediato viene raccolto con la revisione delle prestazioni e dei risultati delle attività operative. Questo feedback proviene dai membri del team, dai clienti o dall'output automatizzato dell'attività. Il feedback immediato viene ricevuto ad esempio dal test A/B e dall'offerta di nuove funzionalità, ed è essenziale per anticipare l'errore (fail fast).

L'analisi retrospettiva viene eseguita regolarmente per acquisire il feedback della revisione dei risultati operativi e dei parametri nel tempo. Queste retrospettive si svolgono alla fine di uno sprint, in base a una cadenza o dopo importanti rilasci o eventi. Questo tipo di ciclo di feedback convalida gli investimenti nelle operazioni o nel carico di lavoro, consente di misurare il successo e comprova la tua strategia.

Risultato desiderato: utilizzi feedback immediato e analisi retrospettiva per apportare miglioramenti. L'applicazione di un meccanismo per acquisire il feedback di utenti e membri del team. L'uso dell'analisi retrospettiva per identificare le tendenze che guidano i miglioramenti.

Anti-pattern comuni:

- Lanci una nuova funzionalità ma non hai modo di ricevere il feedback dei clienti.
- Dopo aver investito in miglioramenti delle operazioni, non conduci una retrospettiva per convalidare gli investimenti.
- Raccogli il feedback dei clienti ma non lo esamini regolarmente.
- I cicli di feedback portano alla proposta di elementi di azione non sono inclusi nel processo di sviluppo software.
- I clienti non ricevono un feedback sui miglioramenti che hanno proposto.

Vantaggi dell'adozione di questa best practice:

- Puoi lavorare a ritroso con il cliente per promuovere nuove funzionalità.
- La cultura della tua organizzazione può reagire più rapidamente ai cambiamenti.
- Le tendenze vengono utilizzate per identificare le opportunità di miglioramento.
- Le retrospettive convalidano gli investimenti effettuati per il carico di lavoro e le operazioni.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

L'implementazione di questa best practice comporta l'utilizzo del feedback immediato e dell'analisi retrospettiva. Questi cicli di feedback promuovono i miglioramenti. Esistono molti meccanismi per il feedback immediato, inclusi questionari, sondaggi dei clienti o moduli di feedback. La tua organizzazione utilizza anche le retrospettive per identificare le opportunità di miglioramento e convalidare le iniziative.

Esempio del cliente

AnyCompany Retail ha creato un modulo web in cui i clienti possono fornire feedback o segnalare problemi. Durante lo scrum settimanale, il feedback degli utenti viene valutato dal team di sviluppo software. Il feedback viene regolarmente utilizzato per guidare l'evoluzione della piattaforma. Viene

eseguita una retrospettiva alla fine di ogni sprint per identificare gli elementi che devono essere migliorati.

Passaggi dell'implementazione

1. Feedback immediato
   - Hai bisogno di un meccanismo per ricevere il feedback dai clienti e dai membri del team. Le attività operative possono anche essere configurate per fornire un feedback automatizzato.
   - L'organizzazione ha bisogno di un processo per rivedere il feedback, determinare cosa migliorare e pianificare il miglioramento.
   - Il feedback deve essere aggiunto al processo di sviluppo software.
   - Quando apporti miglioramenti, contatta l'autore del feedback.
     - Puoi utilizzarlo AWS Systems Manager OpsCenterper creare e tenere traccia di questi miglioramenti come OpsItems.
2. Analisi retrospettiva
   - Conduci le retrospettive alla fine di un ciclo di sviluppo, a una cadenza prestabilita o dopo un rilascio importante.
   - Riunisci le parti interessate coinvolte nel carico di lavoro per la riunione retrospettiva.
   - Crea tre colonne sulla lavagna o in un foglio di lavoro: Fine, Inizio e Mantenimento.
     - Fine riguarda per tutto ciò che vuoi che il team smetta di fare.
     - Inizio riguarda per le idee che vuoi iniziare ad applicare.
     - Mantenimento indica ciò che vuoi continuare a fare.
   - Raccogli il feedback dalle parti interessate.
   - Dai priorità al feedback. Assegna le azioni e le parti interessate a qualsiasi elemento nelle colonne Inizio e Mantenimento.
   - Aggiungi le azioni al processo di sviluppo software e comunica gli aggiornamenti sullo stato alle parti interessate mentre apporti i miglioramenti.

Livello di impegno per il piano di implementazione: medio Per implementare questa best practice è necessario un modo per ricevere il feedback immediato e analizzarlo. Inoltre, è necessario stabilire un processo di analisi retrospettiva.

Risorse

Best practice correlate:

- OPS01-BP01 Valutazione delle esigenze dei clienti esterni: i cicli di feedback sono un meccanismo per raccogliere le esigenze dei clienti esterni.
- OPS01-BP02 Valuta le esigenze interne dei clienti: le parti interessate interne possono utilizzare i cicli di feedback per comunicare necessità e requisiti.
- OPS11-BP02 Eseguire l'analisi post-incidente: le analisi successive agli incidenti sono una forma importante di analisi retrospettiva da condurre dopo gli incidenti.
- OPS11-BP07 Revisioni delle metriche di Perform operations: le revisioni dei parametri operativi identificano tendenze e aree di miglioramento.

Documenti correlati:

- 7 insidie da evitare quando si costruisce un CCOE
- Atlassian Team Playbook - Retrospectives
- Email Definitions: Feedback Loops
- Stabilire cicli di feedback basati sulla revisione del AWS Well-Architected Framework
- IBMGarage Methodology - Organizza una retrospettiva
- Investopedia — Il ciclo PDCS
- Maximizing Developer Effectiveness by Tim Cochran
- White paper su Operations Readiness Reviews (ORR) - Iterazione
- ITILCSI- Miglioramento continuo del servizio
- When Toyota met e-commerce: Lean at Amazon

Video correlati:

- Building Effective Customer Feedback Loops

Esempi correlati:

- Astuto - Open source customer feedback tool
- AWS Soluzioni - Q on nABot AWS
- Fider: una piattaforma per organizzare il feedback dei clienti)

Servizi correlati:

- AWS Systems Manager OpsCenter

OPS11-BP04 Eseguire la gestione della conoscenza

La gestione delle informazioni permette ai membri del team di trovare le informazioni necessarie per svolgere il proprio lavoro. Nelle organizzazioni che promuovono la formazione dei propri dipendenti, le informazioni vengono liberamente condivise, migliorando le competenze personali. Le informazioni possono essere vagliate o cercate. Le informazioni sono accurate e aggiornate. Esistono meccanismi per creare nuove informazioni, aggiornare quelle esistenti e archiviare quelle obsolete. L'esempio più comune di una piattaforma di gestione delle informazioni è un sistema di gestione dei contenuti come un wiki.

Risultato desiderato:

- Accesso per i membri del team a informazioni tempestive e accurate.
- Possibilità di eseguire ricerche nelle informazioni.
- Presenza di un meccanismo per aggiungere, aggiornare e archiviare le informazioni.

Anti-pattern comuni:

- Assenza di un sistema di archiviazione centrale delle informazioni. I membri del team gestiscono i propri appunti su computer locali.
- Presenza di un wiki self-hosted, ma senza alcun meccanismo per la gestione delle informazioni, con informazioni non aggiornate di conseguenza.
- Le informazioni mancanti vengono identificate da qualcuno, ma non esiste un processo per richiederne l'aggiunta nel wiki del team. I dipendenti le aggiungono manualmente ma omettono un passaggio importante, causando un'interruzione.

Vantaggi dell'adozione di questa best practice:

- I membri del team acquisiscono le competenze necessarie perché le informazioni vengono condivise liberamente.
- Nuovi membri del team vengono integrati più facilmente perché la documentazione è aggiornata e può essere oggetto di ricerche.
- Le informazioni sono tempestive, accurate e di utilità pratica.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

La gestione delle informazioni è un aspetto importante delle aziende che promuovono la formazione dei propri dipendenti. Per iniziare, è necessario un repository centrale in cui archiviare le informazioni, un esempio comune del quale è un wiki self-hosted. Devi sviluppare processi per l'aggiunta, l'aggiornamento e l'archiviazione delle informazioni. Sviluppa standard per gli aspetti da documentare e permetti a ciascuno di contribuire.

Esempio del cliente

AnyCompany Retail ospita un Wiki interno in cui sono archiviate tutte le conoscenze. I membri del team sono incoraggiati ad aggiungere il proprio input nella knowledge base durante lo svolgimento delle proprie mansioni quotidiane. Ogni trimestre un team interfunzionale valuta le pagine obsolete e determina se devono essere archiviate o aggiornate.

Passaggi dell'implementazione

1. Per iniziare, identifica il sistema di gestione dei contenuti in cui verranno archiviate le informazioni. Ottieni il consenso delle parti interessate in tutta l'organizzazione.
   a. Se non possiedi un sistema di gestione dei contenuti, valuta se affidarti a un wiki self-hosted o usare un repository con controllo delle versioni come punto di partenza.
2. Sviluppa runbook per l'aggiunta, l'aggiornamento e l'archiviazione delle informazioni. Fornisci ai team la formazione necessaria su questi processi.
3. Identifica le informazioni che devono essere archiviate nel sistema di gestione dei contenuti. Inizia dalle attività quotidiane (runbook e playbook) svolte dai membri del team. Collabora con le parti interessate per classificare in ordine di priorità le informazioni aggiunte.
4. Collabora periodicamente con le parti interessate per identificare out-of-date le informazioni e archiviarle o aggiornarle.

Livello di impegno per il piano di implementazione: medio Se non possiedi un sistema di gestione dei contenuti, puoi configurare un wiki self-hosted o un repository di documenti con controllo delle versioni.

Risorse

Best practice correlate:

- OPS11-BP08 Documentazione e condivisione delle conoscenze acquisite: la gestione delle informazioni semplifica la condivisione delle conclusioni sulle lezioni apprese.

Documenti correlati:

- Atlassian - Knowledge Management

Esempi correlati:

- DokuWiki
- Gollum
- MediaWiki
- Wiki.js

OPS11-BP05 Definizione dei fattori che promuovono il miglioramento

Identifica i fattori che promuovono il miglioramento in modo da valutare e dare priorità alle opportunità sulla base di dati e cicli di feedback. Esplora le opportunità di miglioramento nei sistemi e nei processi e automatizza laddove appropriato.

Risultato desiderato:

- Tieni traccia dei dati provenienti da tutto l'ambiente.
- Esegui la correlazione di eventi e attività ai risultati aziendali.
- Puoi confrontare e contrapporre ambienti e sistemi.
- Mantieni una cronologia dettagliata delle attività relative alle implementazioni e ai risultati.
- Raccogli i dati a supporto del livello di sicurezza.

Anti-pattern comuni:

- Raccogli dati da tutto l'ambiente, ma non correli eventi e attività.
- Raccogli dati dettagliati da tutta la proprietà, aumentando l'attività e i costi di Amazon CloudWatch e AWS CloudTrail, tuttavia non utilizzi questi dati in modo significativo.
- Non tieni conto dei risultati aziendali quando definisci i fattori che promuovono il miglioramento.
- Non misuri gli effetti delle nuove funzionalità.

Vantaggi dell'adozione di questa best practice:

- Determinando i criteri di miglioramento, riduci al minimo l'impatto delle motivazioni basate sugli eventi o degli investimenti influenzati da fattori emotivi.
- Rispondi agli eventi aziendali, non solo a quelli tecnici.
- Misuri l'ambiente per identificare le aree di miglioramento.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

- Comprensione dei fattori che promuovono il miglioramento: è consigliabile apportare modifiche a un sistema solo quando un risultato desiderato è supportato.
  - Funzionalità desiderate: prendi in considerazione le funzionalità e le capacità desiderate quando valuti le opportunità di miglioramento.
    - Novità di AWS
  - Problemi inaccettabili: tieni in considerazione i problemi, i bug e le vulnerabilità inaccettabili quando valuti le opportunità di miglioramento. Tieni traccia delle giuste opzioni di ridimensionamento corretto e individua le opportunità di ottimizzazione.
    - Bollettini sulla sicurezza AWS aggiornati
    - AWS Trusted Advisor
    - Cloud Intelligence Dashboards
  - Requisiti di conformità: quando esamini le opportunità di miglioramento, prendi in considerazione gli aggiornamenti e le modifiche necessarie per mantenere la conformità a normative e policy o per avere diritto al supporto di terze parti.
    - Conformità di AWS
    - Programmi per la conformità di AWS
    - Ultime novità sulla conformità di AWS

Risorse

Best practice correlate:

- OPS01 Priorità dell'organizzazione
- OPS02 Relazioni e proprietà

- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS08 Utilizzare l'osservabilità del carico di lavoro
- OPS09 Comprensione dello stato operativo
- OPS11-BP03 Implementazione di cicli di feedback

Documenti correlati:

- Amazon Athena
- Rapidità
- AWS Conformità di
- Ultime novità sulla conformità di AWS
- Programmi per la conformità di AWS
- AWS Glue
- Bollettini sulla sicurezza AWS aggiornati
- AWS Trusted Advisor
- Export your log data to Amazon S3
- Novità di AWS
- Gli aspetti imprescindibili dell'innovazione orientata al cliente
- Digital Transformation: Hype or a Strategic Necessity?

Video correlati

- AWS re:Invent 2023 - Improve operational efficiency and resilience with Supporto (SUP310)

OPS11-BP06 Validare gli approfondimenti

Rivedi i risultati dell'analisi e le risposte con i team trasversali e i proprietari dell'azienda. Utilizza queste revisioni per definire una visione comune, identificare ulteriori impatti e stabilire le linee d'azione. Adatta le risposte, se necessario.

Risultati desiderati:

- Rivedi regolarmente gli approfondimenti con i proprietari dell'azienda. Gli imprenditori forniscono un contesto aggiuntivo alle informazioni appena acquisite.

- Esamini gli approfondimenti e richiedi il feedback ai colleghi tecnici, quindi condividi le tue conoscenze con i team.
- Pubblichi i dati e gli approfondimenti affinché altri team tecnici e aziendali possano esaminarli. Tieni conto di quanto appreso nelle nuove procedure di altri reparti.
- Riassumi ed esamini i nuovi approfondimenti con i leader senior. I leader senior utilizzano i nuovi approfondimenti per definire la strategia.

Anti-pattern comuni:

- Rilasci una nuova funzionalità che modifica alcuni comportamenti dei clienti. La tua osservabilità non tiene conto di queste modifiche. Non quantifichi i vantaggi di queste modifiche.
- Invii un nuovo aggiornamento e trascuri di aggiornare il tuo. CDN La CDN cache non è più compatibile con l'ultima versione. Misuri la percentuale di richieste con errori. Tutti i tuoi utenti segnalano HTTP 400 errori durante la comunicazione con i server di backend. Analizzi gli errori del cliente e scopri che, avendo misurato la dimensione sbagliata, il tuo tempo è stato improduttivo.
- L'accordo sul livello di servizio prevede un tempo di attività del 99,9% e l'obiettivo del punto di ripristino è di quattro ore. Il proprietario del servizio sostiene che il sistema non subisce tempi di inattività. Implementi una soluzione di replica costosa e complessa, che comporta uno spreco di tempo e denaro.

Vantaggi dell'adozione di questa best practice:

- Convalidando gli approfondimenti con i proprietari dell'azienda e con gli esperti in materia, è possibile stabilire una comprensione comune e gestire il miglioramento in modo più efficace.
- Individui i problemi nascosti e ne tieni conto nelle decisioni future.
- La tua attenzione passa dai risultati tecnici ai risultati aziendali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

- Convalida delle informazioni: interagisci con i responsabili aziendali e gli esperti in materia per garantire la comprensione e l'accordo comuni sul significato dei dati raccolti. Individua ulteriori problemi e impatti potenziali e stabilisci le azioni da intraprendere.

Risorse

Best practice correlate:

- OPS01-BP06 Valuta i compromessi gestendo vantaggi e rischi
- OPS02-BP06 Le responsabilità tra i team sono predefinite o negoziate
- OPS11-BP03 Implementa cicli di feedback

Documenti correlati:

- Progettazione di un centro di eccellenza cloud () CCOE

Video correlati:

- Building observability to increase resiliency

OPS11-BP07 Revisioni delle metriche di Perform operations

Esegui regolarmente un'analisi retrospettiva dei parametri operativi con i partecipanti di vari team da diverse aree dell'azienda. Utilizza queste revisioni per identificare opportunità di miglioramento e potenziali linee d'azione e per condividere le conoscenze acquisite. Cerca opportunità di miglioramento in tutti i tuoi ambienti, ad esempio sviluppo, test e produzione.

Risultato desiderato:

- Esamini di frequente le metriche che hanno un impatto sull'azienda.
- Rilevi ed esamini le anomalie con le tue capacità di osservabilità.
- Utilizzi i dati per supportare i risultati e gli obiettivi aziendali.

Anti-pattern comuni:

- La finestra di manutenzione interrompe un'importante promozione al dettaglio. L'azienda non è al corrente del fatto che i normali interventi di manutenzione possono essere rimandati nel caso vi siano altri eventi di particolare rilievo per l'azienda.
- A causa dell'uso comune di una libreria obsoleta nella tua organizzazione, si è verificata una prolungata interruzione del servizio. In seguito, hai eseguito la migrazione a una libreria supportata. Gli altri team della tua organizzazione non sanno di essere a rischio.

- Non controllate regolarmente i risultati raggiunti dai clienti. SLAs Avete la tendenza a non soddisfare i vostri clienti. SLAs Sono previste sanzioni pecuniarie legate al mancato rispetto del cliente. SLAs

Vantaggi dell'adozione di questa best practice:

- Durante le riunioni che organizzi regolarmente per esaminare le metriche operative, gli eventi e gli incidenti, stabilisci una comprensione comune tra i team.
- Il tuo team si riunisce regolarmente per esaminare metriche e incidenti, il che ti consente di agire sui rischi e riconoscere i clienti. SLAs
- Condividi le lezioni apprese, che forniscono dati per la definizione delle priorità e miglioramenti mirati per ottenere i risultati aziendali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

- Esegui regolarmente un'analisi retrospettiva dei parametri operativi con i partecipanti di vari team da diverse aree dell'azienda.
- Coinvolgi le parti interessate, compresi i team che si occupano di business, sviluppo e operazioni, per convalidare gli esiti del feedback immediato e dall'analisi retrospettiva e per condividere le conoscenze acquisite.
- Utilizza gli approfondimenti di cui dispongono per identificare opportunità di miglioramento e possibili linee d'azione.

Risorse

Best practice correlate:

- OPS08-BP05 Crea dashboard
- OPS09-BP03 Rivedi le metriche operative e dai priorità al miglioramento
- OPS10-BP01 Utilizza un processo per la gestione di eventi, incidenti e problemi

Documenti correlati:

- Amazon CloudWatch

- Riferimento alle CloudWatch metriche e alle dimensioni di Amazon
- Publish custom metrics
- Utilizzo dei CloudWatch parametri di Amazon
- Dashboard e visualizzazioni con CloudWatch

OPS11-BP08 Documentazione e condivisione delle conoscenze acquisite

Documenta e condividi le conoscenze acquisite durante le attività operative per metterle a frutto internamente e nei vari team. La condivisione di quanto appreso dai team comporta maggiori vantaggi all'interno dell'organizzazione. Condividi informazioni e risorse per impedire che si verifichino errori evitabili e semplificare le attività di sviluppo e concentrati sulla distribuzione delle funzionalità desiderate.

Utilizza AWS Identity and Access Management (IAM) per definire le autorizzazioni che consentono un accesso controllato alle risorse che desideri condividere tra i vari account.

Risultato desiderato:

- Utilizzi repository dotati di controllo delle versioni per condividere librerie dell'applicazione, procedure di scripting, documentazione di procedure e altra documentazione di sistema.
- Condividi gli standard dell'infrastruttura come modelli AWS CloudFormation con controllo delle versioni.
- Riesamini le lezioni apprese con i team.

Anti-pattern comuni:

- Per l'uso comune di una libreria contenente degli errori nella tua organizzazione si è verificata una prolungata interruzione del servizio. Successivamente hai eseguito la migrazione a una libreria affidabile. Gli altri team della tua organizzazione non sanno di essere a rischio. Nessuno documenta e condivide l'esperienza relativa a questa libreria e nessuno è consapevole del rischio.
- Hai identificato un caso limite in un microservizio condiviso internamente che causa l'interruzione delle sessioni. Hai aggiornato le chiamate al servizio per evitare questo caso limite. Gli altri team della tua organizzazione non sanno di essere a rischio.
- Hai trovato un modo per ridurre in modo significativo i requisiti di utilizzo della CPU per uno dei tuoi microservizi. Non sai se altri team potrebbero sfruttare questa tecnica.

Vantaggi dell'adozione di questa best practice: condividi le lezioni apprese a supporto del miglioramento e per trarre il massimo vantaggio dall'esperienza.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

- Documenta e condividi le conoscenze acquisite: predisponi procedure per documentare le conoscenze acquisite dall'esecuzione delle attività operative e dalle analisi retrospettive affinché tali informazioni possano essere utilizzate dal altri team.
- Condividi le conoscenze acquisite: predisponi procedure per condividere con tutti i team le conoscenze acquisite e gli artefatti associati. Ad esempio condividi le procedure, le istruzioni, la governance e le best practice aggiornate tramite un wiki accessibile. Condividi script, codice e librerie tramite un repository comune.
  - Sfrutta AWS re:Post Private come servizio informativo per ottimizzare la collaborazione e la condivisione delle conoscenze all'interno dell'organizzazione.

Risorse

Best practice correlate:

- OPS02-BP06 Predefinizione o negoziazione delle responsabilità tra i team
- OPS05-BP01 Utilizzo del controllo delle versioni
- OPS05-BP06 Condivisione degli standard di progettazione
- OPS11-BP03 Implementazione di cicli di feedback
- OPS11-BP07 Revisione dei parametri delle operazioni

Documenti correlati:

- Increase collaboration and securely share cloud knowledge with AWS re:Post Private
- Reduce project delays with a docs-as-code solution

Video correlati:

- AWS re:Invent 2.023 - Collaborate within your company and with AWS using AWS re:Post Private
- Supportos You | Exploring the Incident Management Tabletop Exercise

OPS11-BP09 Dedica tempo per apportare miglioramenti

Dedica tempo e risorse all'interno dei processi per rendere possibile il miglioramento incrementale continuo.

Risultato desiderato:

- Crei duplicati temporanei paralleli di ambienti per ridurre il rischio, lo sforzo e il costo della sperimentazione e dell'esecuzione di test.
- Questi ambienti duplicati possono essere utilizzati per testare le conclusioni di analisi ed esperimenti, ma anche per sviluppare e testare i miglioramenti pianificati.
- Gestisci gamedays e utilizzi Fault Injection Service (FIS) per fornire i controlli e i guardrail necessari ai team per eseguire esperimenti in un ambiente simile alla produzione.

Anti-pattern comuni:

- Si è verificato un problema di prestazioni noto nel server di applicazioni. Il problema viene aggiunto al backlog, dopo l'implementazione prevista delle varie funzionalità. Se la velocità con cui vengono aggiunte le funzionalità pianificate rimane costante, il problema di prestazioni non verrà mai risolto.
- Per supportare il miglioramento continuo, autorizzi amministratori e sviluppatori a utilizzare tutto il loro tempo aggiuntivo per definire e implementare miglioramenti. I miglioramenti non vengono mai completati.
- L'accettazione operativa è stata completata e non si testano più le procedure operative.

Vantaggi dell'adozione di questa best practice: dedicando tempo e risorse all'interno dei processi, puoi rendere possibile il miglioramento incrementale continuo.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

- Allocazione di tempo per apportare miglioramenti: dedica tempo e risorse all'interno dei processi per rendere possibili miglioramenti graduali e continui.
- Implementa modifiche per migliorare e valutare i risultati per favorire il successo.
- Se i risultati non sono in linea con gli obiettivi e il miglioramento resta prioritario, valuta procedure d'azione alternative.

- Simula i carichi di lavoro di produzione durante le giornate di gioco e utilizza le conoscenze conseguite da queste simulazioni per migliorare.

Risorse

Best practice correlate:

- OPS05-BP08 Usa più ambienti

Video correlati:

- AWS re:Invent 2023 - Migliora la resilienza delle applicazioni con il servizio Fault Injection AWS

Sicurezza

Il pilastro della sicurezza contempla la capacità di proteggere dati, sistemi e asset per sfruttare le tecnologie cloud in modo da migliorare la sicurezza. Puoi trovare linee guida prescrittive sull'implementazione nel whitepaper sul pilastro della sicurezza.

Aree delle best practice

- Nozioni di base sulla sicurezza
- Gestione dell’identità e degli accessi
- Rilevamento
- Protezione dell’infrastruttura
- Protezione dei dati
- Risposta agli incidenti
- Sicurezza delle applicazioni

Nozioni di base sulla sicurezza

Domanda

- SEC 1. Come gestire un carico di lavoro in sicurezza?

SEC 1. Come gestire un carico di lavoro in sicurezza?

Per gestire il carico di lavoro in modo sicuro, è necessario applicare le best practice globali a ogni area di sicurezza. Segui i requisiti e i processi definiti in termini di eccellenza operativa a livello organizzativo e di carico di lavoro e applicali a tutte le aree. Rimanere aggiornati con le raccomandazioni di AWS e del settore nonché con l'intelligence sulle minacce aiuta a sviluppare il modello di rischio e gli obiettivi di controllo. L'automazione dei processi di sicurezza, i test e la convalida permettono di dimensionare le operazioni di sicurezza.

Best practice

- SEC01-BP01 Separazione dei carichi di lavoro tramite account
- SEC01-BP02 Utente root e proprietà dell'account sicuro
- SEC01-BP03 Identificazione e convalida degli obiettivi di controllo
- SEC01-BP04 Rimani aggiornato sulle minacce alla sicurezza e sui consigli
- SEC01-BP05 Ridurre l'ambito di gestione della sicurezza
- SEC01-BP06 Automatizza l'implementazione dei controlli di sicurezza standard
- SEC01-BP07 Identificare le minacce e dare priorità alle mitigazioni utilizzando un modello di minaccia.
- SEC01-BP08 Valutazione e implementazione periodiche di nuovi servizi e funzionalità di sicurezza

SEC01-BP01 Separazione dei carichi di lavoro tramite account

Definisci guardrail e isolamento comuni tra ambienti (ad esempio, quelli di produzione, sviluppo e test) e carichi di lavoro mediante una strategia multi-account. La separazione a livello di account è fortemente consigliata, in quanto fornisce un solido confine di isolamento in termini di sicurezza, fatturazione e accesso.

Risultato desiderato: una struttura di account in grado di isolare operazioni cloud, carichi di lavoro non correlati e ambienti in account separati, così da aumentare la sicurezza nell'infrastruttura cloud.

Anti-pattern comuni:

- Inserimento di più carichi di lavoro non correlati con diversi livelli di sensibilità dei dati nello stesso account.
- Scarsa definizione della struttura dell'unità organizzativa (UO).

Vantaggi dell'adozione di questa best practice:

- Riduzione dell'impatto in caso di accesso involontario a un carico di lavoro.
- Governance centralizzata dell'accesso a risorse, regioni e servizi AWS.
- Garanzia di sicurezza dell'infrastruttura cloud grazie a policy e amministrazione centralizzata dei servizi di sicurezza.
- Processo automatizzato di creazione e mantenimento dell'account.
- Audit centralizzati della tua infrastruttura per la conformità e i requisiti normativi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Gli Account AWS offrono un confine di isolamento della sicurezza tra carichi di lavoro o risorse che operano a livelli di sensibilità diversi. AWS fornisce strumenti per gestire i carichi di lavoro del cloud su larga scala attraverso una strategia multi-account per sfruttare questo margine di isolamento. Per linee guida su concetti, modelli e implementazioni di strategie multi-account su AWS, consulta Organizing Your AWS Environment Using Multiple Accounts.

Se disponi di più Account AWS, organizza gli account in una gerarchia definita da livelli di unità organizzative (UO). I controlli di sicurezza possono quindi essere organizzati e applicati alle unità organizzative e agli account membri, stabilendo controlli preventivi coerenti sugli account membri dell'organizzazione. I controlli di sicurezza sono ereditati e consentono di filtrare le autorizzazioni disponibili per gli account membri situati ai livelli inferiori di una gerarchia di unità organizzative. Un buon progetto sfrutta questa ereditarietà per ridurre il numero e la complessità delle policy di sicurezza necessarie per raggiungere i controlli desiderati per ciascun account membro.

È possibile utilizzare due servizi, AWS Organizations e AWS Control Tower, per implementare e gestire questa struttura multi-account nel proprio ambiente AWS. AWS Organizations consente di organizzare gli account in una gerarchia definita da uno o più livelli di unità organizzative, con ciascuna di esse contenente un numero di account membri. Con le policy di controllo dei servizi, l'amministratore dell'organizzazione può stabilire controlli preventivi granulari sugli account membri, mentre AWS Config consente di definire controlli proattivi e investigativi sugli account membri. Molti servizi AWS si integrano con AWS Organizations per offrire controlli amministrativi delegati ed eseguire attività specifiche del servizio su tutti gli account membri dell'organizzazione.

Ripartito nei livelli di AWS Organizations, AWS Control Tower offre una configurazione immediata delle best practice per un ambiente AWS multi-account con una zona di destinazione. La zona di

destinazione è il punto di ingresso nell'ambiente multi-account stabilito da Control Tower. Control Tower offre diversi vantaggi rispetto a AWS Organizations. Tre sono i vantaggi che consentono di migliorare la governance degli account:

- Controlli di sicurezza obbligatori integrati applicati in automatico agli account ammessi nell'organizzazione.
- Controlli opzionali attivabili o disattivabili per un determinato insieme di unità organizzative.
- AWS Control Tower Account Factory consente l'implementazione automatizzata di account contenenti linee di base e opzioni di configurazione preapprovate all'interno della tua organizzazione.

Passaggi dell'implementazione

1. Progettazione di una struttura delle unità organizzative: una struttura delle unità organizzative progettata in modo corretto riduce l'onere di gestione richiesto per creare e mantenere policy di controllo dei servizi e altri controlli di sicurezza. La struttura delle unità organizzative deve essere allineata a esigenze aziendali, sensibilità dei dati e struttura del carico di lavoro.
2. Creazione di una zona di destinazione per il tuo ambiente multi-account: una zona di destinazione costituisce una base infrastrutturale e di sicurezza coerente, che consente all'organizzazione di sviluppare, lanciare e implementare rapidamente carichi di lavoro. Puoi utilizzare una zona di destinazione AWS Control Tower personalizzata per orchestrare il tuo ambiente.
3. Definizione di guardrail: implementa guardrail di sicurezza coerenti per il tuo ambiente mediante la tua zona di destinazione. AWS Control Tower fornisce un elenco di controlli obbligatori e facoltativi implementabili. I controlli obbligatori vengono implementati in automatico in caso di utilizzo di Control Tower. Esamina l'elenco dei controlli altamente consigliati e facoltativi e adotta quelli più adatti alle tue esigenze.
4. Restrizione dell'accesso alle regioni aggiunte di recente: per le nuove Regioni AWS, le risorse IAM, ad esempio utenti e ruoli, verranno propagate solo alle regioni da te specificate. Puoi eseguire questa azione tramite la console in caso di utilizzo di Control Tower o modificando le policy di autorizzazione IAM in AWS Organizations.
5. Presa in esame di AWS CloudFormation StackSets: StackSets consente di implementare risorse, tra cui gruppi, ruoli e policy IAM in vari Account AWS e regioni a partire da un modello approvato.

Risorse

Best practice correlate:

- SEC02-BP04 Fai affidamento su un gestore dell'identità digitale centralizzato

Documenti correlati:

- AWS Control Tower
- Linee guida AWS sugli audit di sicurezza
- Best practice di IAM
- Use CloudFormation StackSets to provision resources across multiple Account AWS and regions
- Organizations FAQ
- AWS Organizations terminology and concepts
- Best Practices for Service Control Policies in an AWS Organizations Multi-Account Environment
- AWS Account Management Reference Guide
- Organizzazione dell'ambiente AWS che utilizza più account

Video correlati:

- Organizing Your AWS Environment Using Multiple Accounts
- Security Best Practices the Well-Architected Way
- Building and Governing Multiple Accounts using AWS Control Tower
- Enable Control Tower for Existing Organizations

SEC01-BP02 Utente root e proprietà dell'account sicuro

L'utente root è la figura più privilegiata di un Account AWS, ha pieno accesso amministrativo a tutte le risorse dell'account e, in alcuni casi, non può essere limitato dalle policy di sicurezza. Disattivare l'accesso programmatico all'utente root, stabilire controlli appropriati per l'utente root ed evitare l'uso di routine dell'utente root aiuta a ridurre il rischio di esposizione involontaria delle credenziali root e la conseguente compromissione dell'ambiente cloud.

Risultato desiderato: proteggere l'utente root riduce la possibilità di danni accidentali o intenzionali dovuti all'uso improprio delle credenziali dell'utente root. La creazione di controlli investigativi può anche permettere di avvisare il personale appropriato quando vengono eseguite azioni utilizzando l'utente root.

Anti-pattern comuni:

- Utilizzo dell'utente root per attività diverse da quelle che richiedono le proprie credenziali.
- Nessun test dei piani di emergenza su base regolare per verificare il funzionamento di infrastrutture critiche, processi e personale durante un'emergenza.
- Analisi limitata al tipico flusso di accesso all'account, trascurando di considerare o testare metodi alternativi di ripristino dell'account.
- Nessuna gestione di DNS, server di posta elettronica e provider telefonici come parte del perimetro di sicurezza critico, in quanto utilizzati nel flusso di recupero degli account.

Vantaggi dell'adozione di questa best practice: proteggere l'accesso all'utente root aumenta la sicurezza circa controlli e audit delle azioni nell'account

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

AWS offre molti strumenti per proteggere gli account. Tuttavia, poiché alcune di queste misure non sono attivate per impostazione predefinita, è necessario intervenire direttamente per implementarle. Queste raccomandazioni costituiscono i passi fondamentali per mettere in sicurezza il proprio Account AWS. Durante l'implementazione di questi passaggi, è importante creare un processo di valutazione e monitoraggio continuo dei controlli di sicurezza.

La prima creazione di un Account AWS parte con una singola identità che ha accesso completo a tutti i servizi e risorse AWS presenti nell'account. Questa identità è chiamata utente root dell'Account AWS. Puoi accedere come utente root utilizzando l'indirizzo e-mail e la password usati per creare l'account. A causa dell'accesso elevato concesso all'utente root AWS, è necessario limitare l'uso dell'utente root AWS all'esecuzione di attività che lo richiedano nello specifico. Le credenziali di accesso dell'utente root devono essere tenute sotto stretta sorveglianza e l'autenticazione a più fattori (MFA) deve essere sempre utilizzata per l'utente root dell'Account AWS.

Oltre al normale flusso di autenticazione per accedere all'utente root utilizzando un nome utente, una password e un dispositivo di autenticazione a più fattori (MFA), esistono flussi di recupero dell'account che consentono di accedere all'utente root dell'Account AWS grazie all'accesso all'indirizzo e-mail e al numero di telefono associati all'account. Pertanto, è altrettanto importante proteggere l'account e-mail dell'utente root a cui vengono inviati l'e-mail di recupero e il numero di telefono associato all'account. Prendi anche in considerazione le potenziali dipendenze circolari, quando l'indirizzo e-mail associato all'utente root è ospitato su server di posta elettronica o su risorse del servizio dei nomi di dominio (DNS) dello stesso Account AWS.

Quando si utilizza AWS Organizations, esistono più Account AWS, ciascuno con un utente root. Un account è designato come account di gestione e sotto l'account di gestione è possibile aggiungere diversi livelli di account membri. La priorità è proteggere l'utente root dell'account di gestione, quindi occuparsi degli utenti root degli account membri. La strategia per la protezione dell'utente root dell'account di gestione può essere diversa da quella degli utenti root degli account membri ed è possibile effettuare controlli di sicurezza preventivi sugli utenti root degli account membri.

Passaggi dell'implementazione

Per stabilire i controlli per l'utente root, si consigliano i seguenti passaggi di implementazione. Ove applicabile, le raccomandazioni devono essere confrontate con la versione 1.4.0 del benchmark CIS AWS Foundations. Oltre a questi passaggi, consulta le linee guida sulle best practice AWS per proteggere il tuo account Account AWS e le tue risorse.

Controlli preventivi

1. Imposta informazioni di contatto precise per l'account.
    a. Queste informazioni vengono utilizzate per il flusso di recupero della password persa, per il flusso di recupero dell'account del dispositivo MFA perso e per le comunicazioni critiche relative alla sicurezza con il team.
    b. Utilizza un indirizzo e-mail ospitato dal dominio aziendale, preferibilmente una lista di distribuzione, come indirizzo e-mail dell'utente root. L'utilizzo di una lista di distribuzione anziché l'account e-mail di un singolo individuo offre una maggiore ridondanza e continuità di accesso all'account root per lunghi periodi di tempo.
    c. Il numero di telefono indicato nelle informazioni di contatto deve essere dedicato e sicuro per questo scopo. Il numero di telefono non deve essere indicato o condiviso con nessuno.
2. Non creare chiavi di accesso per l'utente root. Se sono presenti chiavi di accesso, rimuovile (CIS 1.4).
    a. Elimina le credenziali programmatiche a lunga durata (chiavi di accesso e segrete) per l'utente root.
    b. Se esistono già chiavi di accesso dell'utente root, fai in modo che i processi che utilizzano tali chiavi passino all'utilizzo di chiavi di accesso temporanee provenienti da un ruolo AWS Identity and Access Management (IAM), quindi elimina le chiavi di accesso dell'utente root.
3. Stabilisci se è necessario memorizzare le credenziali per l'utente root.
    a. Se utilizzi AWS Organizations per creare nuovi account membri, la password iniziale dell'utente root sui nuovi account membro è impostata su un valore casuale che non è visibile a te. Prendi

in considerazione l'utilizzo del flusso di reimpostazione della password del tuo account di gestione AWS Organization per accedere all'account membro, se necessario.

b. Per gli Account AWS standalone o per l'account di gestione di AWS Organization, considera la creazione e l'archiviazione sicura delle credenziali per l'utente root. Usa MFA per l'utente root.

4. Usa i controlli preventivi per gli utenti root degli account membri in ambienti AWS multi-account.

a. Prendi in considerazione l'utilizzo del guardrail preventivo Disallow Creation of Root Access Keys for Root User per gli account membri.

b. Prendi in considerazione l'utilizzo del guardrail preventivo Disallow Actions as a Root User per gli account membri.

5. Se sono necessarie le credenziali per l'utente root:

a. Utilizza una password complessa.

b. Attiva l'autenticazione a più fattori (MFA) per l'utente root, in particolare per gli account dei manager (paganti) AWS Organizations (CIS 1.5).

c. Prendi in considerazione i dispositivi MFA hardware per la resilienza e la sicurezza, in quanto i dispositivi monouso possono ridurre le possibilità che i dispositivi contenenti i codici MFA vengano riutilizzati per altri scopi. Verifica che i dispositivi hardware MFA alimentati da una batteria siano sostituiti regolarmente. (CIS 1.6)

- Per configurare l'MFA per l'utente root, segui le istruzioni per creare un dispositivo MFA virtuale o un dispositivo MFA hardware.

d. Prendi in considerazione la registrazione di più dispositivi MFA per il backup. Sono consentiti fino a 8 dispositivi MFA per account.

- Tieni presente che la registrazione di più di un dispositivo MFA per l'utente root disattiva in automatico il flusso per il recupero dell'account in caso di smarrimento del dispositivo MFA.

e. Conserva la password in modo sicuro e considera le dipendenze circolari se la password viene conservata elettronicamente. Non memorizzare la password in modo tale da richiedere l'accesso allo stesso Account AWS per ottenerla.

6. Facoltativo: valuta la possibilità di stabilire un programma di rotazione periodica delle password per l'utente root.

- Le best practice per la gestione delle credenziali dipendono dai requisiti normativi e di policy. Gli utenti root protetti da MFA non dipendono dalla password come unico fattore di autenticazione.
- La modifica periodica della password dell'utente root riduce il rischio di utilizzo improprio di una password esposta inavvertitamente.

Controlli di rilevamento

- Crea allarmi per rilevare l'uso delle credenziali root (CIS 1.7). Amazon GuardDuty può monitorare e inviare avvisi sull'utilizzo delle credenziali API dell'utente root tramite l'esito RootCredentialUsage.
- Valuta e implementa i controlli investigativi inclusi nel pacchetto di conformità del pilastro della sicurezza AWS Well-Architected per AWS Config o, in caso di utilizzo di AWS Control Tower, i controlli fortemente consigliati disponibili in Control Tower.

Guida operativa

- Stabilisci chi nell'organizzazione deve avere accesso alle credenziali dell'utente root.
  - Utilizza una regola a due persone, in modo che nessun individuo abbia accesso a tutte le credenziali necessarie e all'MFA per ottenere l'accesso come utente root.
  - Verifica che l'organizzazione, e non un singolo individuo, mantenga il controllo sul numero di telefono e sull'alias e-mail associati all'account (utilizzati per il ripristino della password e il flusso di ripristino MFA).
- Utilizza l'utente root solo in via eccezionale (CIS 1.7).
  - L'utente root AWS non deve essere utilizzato per le attività giornaliere e nemmeno per quelle amministrative. Effettua l'accesso come utente root solo per eseguire attività AWS che richiedono l'utente root. Tutte le altre azioni devono essere eseguite da altri utenti che assumono i ruoli appropriati.
- Verifica periodicamente che l'accesso all'utente root sia funzionante, in modo da testare le procedure prima di una situazione di emergenza che richieda l'uso delle credenziali dell'utente root.
- Verifica a intervalli regolari il funzionamento dell'indirizzo e-mail associato all'account e quelli indicati nei contatti alternativi. Monitora queste caselle di posta elettronica per le notifiche di sicurezza che potresti ricevere da <abuse@amazon.com>. Assicurati inoltre che i numeri di telefono associati all'account siano attivi.
- Prepara procedure di risposta agli incidenti per rispondere all'uso improprio dell'account root. Consulta la AWS Security Incident Response Guide e le best practice nella sezione Risposta agli imprevisti del whitepaper sul pilastro della sicurezza per ulteriori informazioni circa la creazione di una strategia di risposta agli incidenti adatta al tuo Account AWS.

Risorse

Best practice correlate:

- SEC01-BP01 Separazione dei carichi di lavoro tramite account
- SEC02-BP01 Utilizzo di meccanismi di accesso efficaci
- SEC03-BP02 Concessione dell'accesso con privilegio minimo
- SEC03-BP03 Determinazione di un processo per l'accesso di emergenza
- SEC10-BP05 Preassegnazione dell'accesso

Documenti correlati:

- AWS Control Tower
- Linee guida AWS sugli audit di sicurezza
- Best practice di IAM
- Amazon GuardDuty: avviso di utilizzo delle credenziali root
- Step-by-step guidance on monitoring for root credential use through CloudTrail
- MFA tokens approved for use with AWS
- Implementazione di break glass access su AWS
- Top 10 security items to improve in your Account AWS
- What do I do if I notice unauthorized activity in my Account AWS?

Video correlati:

- Organizing Your AWS Environment Using Multiple Accounts
- Security Best Practices the Well-Architected Way
- Limiting use of AWS root credentials from AWS re:inforce 2022 – Security best practices with AWS IAM

SEC01-BP03 Identificazione e convalida degli obiettivi di controllo

In base ai requisiti di conformità e ai rischi identificati dal modello di rischio, individua e convalida gli obiettivi di controllo e i controlli da applicare al carico di lavoro. La convalida continua degli obiettivi di controllo e dei controlli aiuta a misurare l'efficacia della mitigazione dei rischi.

Risultato desiderato: gli obiettivi di controllo della sicurezza della tua azienda sono ben definiti e in linea con i requisiti di conformità. I controlli vengono implementati e applicati attraverso l'automazione

e le policy e vengono costantemente valutati per verificarne l'efficacia nel raggiungimento degli obiettivi. Le prove dell'efficacia, sia in un determinato momento che in un determinato periodo di tempo, sono prontamente comunicate ai revisori.

Anti-pattern comuni:

- I requisiti normativi, le aspettative del mercato e gli standard di settore per una sicurezza certa non sono ben compresi dalla tua azienda.
- I framework di sicurezza informatica e gli obiettivi di controllo non sono allineati ai requisiti dell'azienda.
- L'implementazione dei controlli non è perfettamente allineata agli obiettivi di controllo in modo misurabile.
- L'automazione non viene utilizzata per creare report sull'efficacia dei tuoi controlli.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I framework di sicurezza informatica comunemente utilizzati sono molti e possono costituire la base per gli obiettivi di controllo della sicurezza. Per determinare quale sia il framework più adatto alle tue esigenze, considera i requisiti normativi, le aspettative del mercato e gli standard di settore dell'azienda. Tra gli esempi citiamo AICPA SOC 2, HITRUST, PCI-DSS, ISO 27.001 e NIST SP 800-53.

Per gli obiettivi di controllo identificati, occorre comprendere in che modo i servizi AWS utilizzati permettono di conseguirli. Utilizza AWS Artifact per individuare documentazione e report in linea con i tuoi framework di riferimento, che illustrino l'ambito di responsabilità coperto da AWS e linee guida per il restante ambito di tua responsabilità. Per ulteriori indicazioni specifiche sui servizi in linea con le varie dichiarazioni di controllo del framework, consulta le AWS Customer Compliance Guides.

Nel definire i controlli che raggiungono i tuoi obiettivi, codifica l'applicazione utilizzando i controlli preventivi e automatizza le mitigazioni mediante i controlli di rilevamento. Aiuta a prevenire configurazioni e azioni delle risorse non conformi su AWS Organizations mediante le policy di controllo dei servizi (SCP). Implementa le regole in AWS Config al fine di monitorare e segnalare le risorse non conformi, quindi passa a un modello di applicazione delle regole una volta che sei sicuro del loro comportamento. Per implementare set di regole predefinite e gestite in linea con i tuoi framework di sicurezza informatica, prendi in considerazione l'uso degli standard AWS Security Hub CSPM come prima opzione. Lo standard AWS Foundational Service Best Practices (FSBP) e il

CIS AWS Foundations Benchmark sono validi punti di partenza con controlli che si allineano a molti obiettivi condivisi da più framework standard. Se Security Hub CSPM non dispone a livello intrinseco dei rilevamenti di controllo desiderati, è possibile integrarlo mediante i pacchetti di conformità AWS Config.

Utilizza i bundle dei partner APN consigliati dal team AWS Global Security and Compliance Acceleration (GSCA) per ottenere assistenza da consulenti di sicurezza, agenzie di consulenza, sistemi di raccolta e di reporting delle prove, revisori e altri servizi complementari, se necessario.

Passaggi dell'implementazione

1. Valuta i framework di sicurezza informatica comuni e allinea i tuoi obiettivi di controllo a quelli scelti.
2. Ottieni la documentazione pertinente sulle linee guida e le responsabilità per il tuo framework utilizzando AWS Artifact. Comprendi quali parti della conformità rientrano nel modello di responsabilità condivisa AWS e quali sono di tua competenza.
3. Utilizza le policy di controllo dei servizi, le policy sulle risorse, le policy di attendibilità dei ruoli e altri guardrail per prevenire configurazioni e azioni delle risorse non conformi.
4. Valuta l'implementazione di standard Security Hub CSPM e pacchetti di conformità AWS Config in linea con i tuoi obiettivi di controllo.

Risorse

Best practice correlate:

- SEC03-BP01 Definizione dei requisiti di accesso
- SEC04-BP01 Configurazione dei log di servizi e applicazioni
- SEC07-BP01 Comprendere lo schema di classificazione dei dati
- OPS01-BP03 Valutazione dei requisiti di governance
- OPS01-BP04 Valutazione dei requisiti di conformità
- PERF01-BP05 Uso delle policy e delle architetture di riferimento
- COST02-BP01 Sviluppo di policy basate sui requisiti dell'organizzazione

Documenti correlati:

- AWS Customer Compliance Guides

Strumenti correlati:

- AWS Artifact

SEC01-BP04 Rimani aggiornato sulle minacce alla sicurezza e sui consigli

Rimani aggiornato sulle minacce più recenti e sulle misure di mitigazione monitorando le pubblicazioni di intelligence sulle minacce del settore e i feed di dati per gli aggiornamenti. Valuta le offerte di servizi gestiti che si aggiornano in automatico in base ai dati sulle minacce più recenti.

Risultato desiderato: rimani informato mentre le pubblicazioni di settore si aggiornano con le ultime minacce e raccomandazioni. L'automazione viene utilizzata per rilevare potenziali vulnerabilità ed esposizioni man mano che si identificano nuove minacce. Intraprendi azioni di mitigazione contro queste minacce. Adottate AWS servizi che si aggiornano automaticamente con le più recenti informazioni sulle minacce.

Anti-pattern comuni:

- Non disporre di un meccanismo affidabile e ripetibile per rimanere informati sulle ultime informazioni sulle minacce.
- Mantenere un inventario manuale del portafoglio tecnologico, dei carichi di lavoro e delle dipendenze che richiedono un esame umano per individuare potenziali vulnerabilità ed esposizioni.
- Non disporre di meccanismi per aggiornare i carichi di lavoro e le dipendenze alle ultime versioni disponibili, che forniscono mitigazioni note delle minacce.

Vantaggi dell'adozione di questa best practice: l'utilizzo di fonti di intelligence sulle minacce per rimanere aggiornati riduce il rischio di lasciarsi sfuggire importanti cambiamenti nel panorama delle minacce in grado di pregiudicare la tua azienda. L'automazione in atto per scansionare, rilevare e correggere eventuali vulnerabilità o esposizioni nei carichi di lavoro e nelle relative dipendenze può aiutarti a mitigare i rischi in modo rapido e prevedibile, rispetto alle alternative manuali. In questo modo puoi controllare i tempi e i costi relativi alla mitigazione delle vulnerabilità.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Consulta le pubblicazioni di intelligence sulle minacce per costanti aggiornamenti sul panorama delle minacce. Consultate la knowledge base MITREATT&CK per la documentazione sulle tattiche, le

tecniche e le procedure antagonistiche note (). TTPs Consulta MITRE l'elenco delle vulnerabilità e delle esposizioni comuni (CVE) per rimanere informato sulle vulnerabilità note nei prodotti su cui fai affidamento. Comprendi i rischi critici per le applicazioni web con il popolare OWASP progetto Top 10 dell'Open Worldwide Application Security Project (OWASP).

Rimani aggiornato sugli eventi di AWS sicurezza e sulle procedure di correzione consigliate con AWS Security Bulletins for. CVEs

Per ridurre l'impegno complessivo e il sovraccarico legati all'aggiornamento, prendi in considerazione l'utilizzo di AWS servizi che incorporano automaticamente nuove informazioni sulle minacce nel tempo. Ad esempio, Amazon si GuardDuty aggiorna con le informazioni sulle minacce del settore per rilevare comportamenti anomali e firme di minacce all'interno dei tuoi account. Amazon Inspector mantiene automaticamente aggiornato un database dei CVEs dati utilizzati per le sue funzionalità di scansione continua. AWS WAF e AWS Shield Advanced forniscono gruppi di regole gestiti, aggiornati in automatico all'emergere di nuove minacce.

Esamina il pilastro dell'eccellenza operativa Well-Architected per la gestione e l'applicazione di patch automatizzate del parco.

Passaggi dell'implementazione

- Abbonati agli aggiornamenti per le pubblicazioni di intelligence sulle minacce pertinenti alla tua azienda e al tuo settore. Abbonati ai bollettini sulla sicurezza AWS .
- Prendi in considerazione l'adozione di servizi che incorporano automaticamente nuove informazioni sulle minacce, come Amazon GuardDuty e Amazon Inspector.
- Implementa una strategia di gestione e applicazione delle patch del parco in linea con le best practice del pilastro dell'eccellenza operativa Well-Architected.

Risorse

Best practice correlate:

- SEC01-BP07 Identifica le minacce e dai priorità alle mitigazioni utilizzando un modello di minaccia
- OPS01-BP05 Valuta il panorama delle minacce
- OPS11-BP01 Adottate un processo per il miglioramento continuo

SEC01-BP05 Ridurre l'ambito di gestione della sicurezza

Determina se puoi ridurre l'ambito di sicurezza utilizzando AWS servizi che spostano la gestione di determinati controlli su AWS (servizi gestiti). Questi servizi possono contribuire a ridurre le attività di manutenzione della sicurezza, come il provisioning dell'infrastruttura, l'impostazione del software, il patching o i backup.

Risultato desiderato: quando si selezionano i AWS servizi per il carico di lavoro, si considera l'ambito della gestione della sicurezza. Il costo delle spese generali di gestione e delle attività di manutenzione (il costo totale di proprietà oTCO) viene confrontato con il costo dei servizi selezionati, oltre ad altre considerazioni di Well-Architected. La documentazione relativa al AWS controllo e alla conformità viene incorporata nelle procedure di valutazione e verifica del controllo.

Anti-pattern comuni:

- Implementazione dei carichi di lavoro senza comprendere a fondo il modello di responsabilità condivisa per i servizi selezionati.
- Hosting di database e altre tecnologie su macchine virtuali senza aver valutato un servizio gestito equivalente.
- Mancata inclusione delle attività di gestione della sicurezza nel costo totale di proprietà delle tecnologie di hosting su macchine virtuali rispetto alle opzioni di servizio gestito.

Vantaggi dell'adozione di questa best practice: l'utilizzo di servizi gestiti può ridurre l'onere complessivo della gestione dei controlli operativi della sicurezza, così da ridurre rischi per la sicurezza e costo totale di proprietà. Il tempo che altrimenti sarebbe dedicato a determinate attività di sicurezza può essere reinvestito in attività che forniscono maggior valore alla tua azienda. I servizi gestiti possono anche ridurre l'ambito dei requisiti di conformità spostando alcuni requisiti di controllo su AWS.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Le modalità di integrazione dei componenti del carico di lavoro su AWS sono molteplici. L'installazione e l'esecuzione di tecnologie su EC2 istanze Amazon spesso richiedono l'assunzione della maggior parte della responsabilità complessiva in materia di sicurezza. Per contribuire a ridurre l'onere derivante dall'utilizzo di determinati controlli, AWS individua i servizi gestiti che riducano la portata del modello di responsabilità condivisa e comprendi come utilizzarli nell'architettura

esistente. Gli esempi includono l'utilizzo di Amazon Relational Database Service (RDSAmazon) per la distribuzione di database, Amazon Elastic Kubernetes Service (Amazon) o Amazon Elastic ECS Container EKS Service(Amazon) per orchestrare contenitori o l'utilizzo di opzioni serverless. Quando sviluppi nuove applicazioni, pensa a quali servizi possono contribuire a ridurre i tempi e i costi di implementazione e gestione dei controlli di sicurezza.

Anche i requisiti di conformità possono essere un fattore di scelta dei servizi. I servizi gestiti possono spostare la conformità di alcuni requisiti a. AWS Parlate con il vostro team addetto alla conformità in merito alla loro capacità di controllare gli aspetti dei servizi che gestite e gestite e di accettare le dichiarazioni di controllo nei rapporti di AWS audit pertinenti. Potete fornire agli auditor o AWS Artifactalle autorità di regolamentazione gli elementi degli audit presenti come prova dei controlli di sicurezza. AWS Puoi anche utilizzare le linee guida sulla responsabilità fornite da alcuni degli elementi di AWS audit per progettare la tua architettura, insieme alle Customer Compliance Guides.AWS Queste indicazioni aiutano a determinare i controlli di sicurezza aggiuntivi da mettere in atto per supportare i casi d'uso specifici del sistema.

Quando utilizzi servizi gestiti, acquisisci familiarità con il processo di aggiornamento delle risorse alle versioni più recenti (ad esempio, l'aggiornamento della versione di un database gestito da Amazon RDS o il runtime di un linguaggio di programmazione per una AWS Lambda funzione). Anche se il servizio gestito può eseguire questa operazione per tuo conto, la configurazione della tempistica dell'aggiornamento e la conoscenza dell'impatto sulle tue operazioni restano di tua responsabilità. Strumenti come AWS Health ti consentono di tracciare e gestire questi aggiornamenti in tutti i tuoi ambienti.

Passaggi dell'implementazione

1. Valuta i componenti del tuo carico di lavoro sostituibili con un servizio gestito.
    a. Se stai migrando un carico di lavoro verso AWS, prendi in considerazione la riduzione della gestione (tempo e spese) e la riduzione del rischio quando valuti se riospitare, rifattorizzare, ripiattaforma, ricostruire o sostituire il carico di lavoro. A volte un investimento aggiuntivo all'inizio di una migrazione può comportare risparmi significativi nel lungo periodo.
2. Prendi in considerazione l'implementazione di servizi gestitiRDS, come Amazon, anziché installare e gestire le tue implementazioni tecnologiche.
3. Utilizza le linee guida sulla responsabilità riportate AWS Artifact di seguito per determinare i controlli di sicurezza da adottare per il tuo carico di lavoro.
4. Tenete un inventario delle risorse in uso e continuate a up-to-date utilizzare nuovi servizi e approcci per identificare nuove opportunità per ridurre l'ambito di applicazione.

Risorse

Best practice correlate:

- PERF02-BP01 Seleziona le migliori opzioni di elaborazione per il tuo carico di lavoro
- PERF03-BP01 Utilizza un data store appositamente progettato che supporti al meglio i requisiti di accesso e archiviazione dei dati
- SUS05-BP03 Utilizza servizi gestiti

Documenti correlati:

- Eventi del ciclo di vita pianificati per AWS Health

Strumenti correlati:

- AWS Health
- AWS Artifact
- AWS Customer Compliance Guides

Video correlati:

- Come posso migrare a un'istanza Amazon RDS o Aurora SQL My DB utilizzando? AWS DMS
- AWS re:Invent 2023 - Gestisci gli eventi del ciclo di vita delle risorse su larga scala con AWS Health

SEC01-BP06 Automatizza l'implementazione dei controlli di sicurezza standard

Applica DevOps pratiche moderne mentre sviluppi e distribuisci controlli di sicurezza standard in tutti i tuoi ambienti. AWS Definisci controlli e configurazioni di sicurezza standard utilizzando modelli Infrastructure as Code (IaC), acquisisci le modifiche in un sistema di controllo delle versioni, testa le modifiche come parte di una pipeline CI/CD e automatizza l'implementazione delle modifiche ai tuoi ambienti. AWS

Risultato desiderato: i modelli IaC acquisiscono controlli di sicurezza standardizzati, inserendoli in un sistema di controllo delle versioni. Le pipeline CI/CD consentono di rilevare le modifiche e automatizzare i test e l'implementazione degli ambienti. AWS Sono presenti guardrail per rilevare e fornire avvisi in caso di configurazioni errate nei modelli prima di procedere all'implementazione. I

carichi di lavoro vengono implementati in ambienti dotati di controlli standard. I team hanno accesso all'implementazione di configurazioni di servizio approvate tramite un meccanismo self-service. Sono disponibili strategie di backup e ripristino sicure per le configurazioni di controllo, gli script e i dati correlati.

Anti-pattern comuni:

- Apportare modifiche ai controlli di sicurezza standard manualmente, tramite una console Web o un'interfaccia a riga di comando.
- Affidarsi ai singoli team del carico di lavoro per implementare manualmente i controlli definiti da un team centrale.
- Affidarsi a un team di sicurezza centrale per implementare i controlli a livello di carico di lavoro su richiesta di un team del carico di lavoro.
- Consentire agli stessi individui o team di sviluppare, testare e implementare script di automazione per il controllo della sicurezza senza un'adeguata separazione dei compiti o dei controlli e degli equilibri.

Vantaggi dell'adozione di questa best practice: l'utilizzo di modelli per definire i controlli di sicurezza standard consente di tracciare e confrontare le modifiche nel tempo con un sistema di controllo delle versioni. L'uso dell'automazione per testare e implementare le modifiche crea standardizzazione e prevedibilità, aumentando le possibilità di una corretta implementazione e riducendo le attività manuali ripetitive. Fornire un meccanismo self-service per consentire ai team addetti al carico di lavoro di implementare servizi e configurazioni approvati riduce il rischio di configurazioni errate e usi impropri. Questo li aiuta anche a incorporare i controlli nelle prime fasi del processo di sviluppo.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Se si seguono le pratiche descritte in SEC01-BP01 Separazione dei carichi di lavoro utilizzando gli account, si ottengono più unità per ambienti diversi da gestire. Account AWS AWS Organizations Sebbene ciascuno di questi ambienti e carichi di lavoro possa richiedere controlli di sicurezza distinti, puoi standardizzarne alcuni in tutta l'organizzazione. Gli esempi includono l'integrazione di gestori dell'identità digitale centralizzati, la definizione di reti e firewall e la configurazione di posizioni standard per l'archiviazione e l'analisi dei log. Allo stesso modo in cui puoi utilizzare infrastructure as code (IaC) per applicare lo stesso criterio dello sviluppo del codice dell'applicazione al provisioning dell'infrastruttura, puoi usare l'IaC anche per definire e implementare controlli di sicurezza standard.

Se possibile, definisci i controlli di sicurezza in modo dichiarativo, ad esempio in AWS CloudFormation, e archiviali in un sistema di controllo del codice sorgente. Utilizza DevOps procedure per automatizzare l'implementazione dei controlli per versioni più prevedibili, esegui test automatici utilizzando strumenti come AWS CloudFormation Guarde rilevando eventuali differenze tra i controlli implementati e la configurazione desiderata. Puoi utilizzare servizi come AWS CodePipeline, AWS CodeBuild e AWS CodeDeploy per creare una pipeline CI/CD. Prendi in considerazione le indicazioni contenute nella sezione Organizzazione AWS dell'ambiente utilizzando più account per configurare questi servizi nei rispettivi account, separati dalle altre pipeline di distribuzione.

È inoltre possibile definire modelli per standardizzare la definizione e la distribuzione Account AWS, i servizi e le configurazioni. Questa tecnica consente a un team di sicurezza centrale di gestire queste definizioni e di fornirle ai team che si occupano dei carichi di lavoro attraverso un approccio self-service. Un modo per raggiungere questo obiettivo è utilizzare Service Catalog, dove è possibile pubblicare modelli come prodotti che i team addetti al carico di lavoro possono integrare nelle proprie implementazioni della pipeline. AWS Control Tower offre alcuni modelli e controlli come punto di partenza. Control Tower offre anche la funzionalità Account Factory, che consente ai team addetti al carico di lavoro di creare di nuovi Account AWS mediante gli standard definiti da te. Questa funzionalità aiuta a rimuovere le dipendenze da un team centrale per l'approvazione e la creazione di nuovi account quando vengono identificati come necessari dai team del carico di lavoro. Potresti aver bisogno di questi account per isolare i diversi componenti del carico di lavoro in base a motivi quali la funzione che svolgono, la sensibilità dei dati elaborati o il loro comportamento.

Passaggi dell'implementazione

1. Determina come archivierai e manterrai i tuoi modelli in un sistema di controllo delle versioni.
2. Crea pipeline CI/CD per testare e implementare i tuoi modelli. Definisci i test per verificare che non ci siano configurazioni errate e che i modelli siano conformi agli standard aziendali.
3. Crea un catalogo di modelli standardizzati da distribuire ai team addetti ai carichi di lavoro Account AWS e di servizi in base alle tue esigenze.
4. Implementa strategie di backup e ripristino sicure per le configurazioni di controllo, gli script e i dati correlati.

Risorse

Best practice correlate:

- OPS05-BP01 Usa il controllo della versione

- OPS05-BP04 Utilizza sistemi di gestione della compilazione e dell'implementazione
- REL08-BP05 Implementa le modifiche con l'automazione
- SUS06-BP01 Adotta metodi in grado di introdurre rapidamente miglioramenti alla sostenibilità

Documenti correlati:

- Organizzazione dell'ambiente utilizzando più account AWS

Esempi correlati:

- Automatizza la creazione di account e il provisioning delle risorse utilizzando Service Catalog e AWS OrganizationsAWS Lambda
- Rafforza la DevOps pipeline e proteggi i dati con Gestione dei segreti AWS, e AWS KMSAWS Certificate Manager

Strumenti correlati:

- AWS CloudFormation Guard
- Landing Zone Accelerator attivo AWS

SEC01-BP07 Identificare le minacce e dare priorità alle mitigazioni utilizzando un modello di minaccia.

Effettua la modellazione delle minacce per identificare e mantenere un registro aggiornato delle minacce potenziali e delle relative mitigazioni per il carico di lavoro. Definisci le priorità delle minacce e adatta le mitigazioni dei controlli di sicurezza per prevenire, intercettare e rispondere. Riesamina e mantieni questo aspetto nel contesto del tuo carico di lavoro e dell'evoluzione del panorama della sicurezza.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Che cos'è la modellazione delle minacce?

"La modellazione delle minacce mira a identificare, comunicare e comprendere minacce e mitigazioni nel contesto della protezione di qualcosa di valore". – The Open Web Application Security Project (OWASP) Application Threat Modeling

Perché adottare la modellazione delle minacce?

I sistemi sono complessi, e nel tempo lo diventano sempre di più, e capaci di fornire un maggiore valore aziendale e una maggiore soddisfazione e coinvolgimento dei clienti. Ciò significa che le decisioni di progettazione IT devono tenere conto di un numero sempre maggiore di casi d'uso. Questa complessità e il numero di combinazioni di casi d'uso rendono in genere gli approcci non strutturati inefficaci per individuare e mitigare le minacce. È invece necessario un approccio sistematico per enumerare le potenziali minacce al sistema ed elaborare le mitigazioni, oltre che per stabilirne le priorità per assicurarsi che le risorse limitate dell'organizzazione abbiano il massimo impatto nel migliorare lo stato di sicurezza complessiva del sistema.

La modellazione delle minacce è progettata per offrire questo approccio sistematico, con l'obiettivo di trovare e affrontare i problemi nelle prime fasi del processo di progettazione, quando le mitigazioni hanno un costo e un impegno relativi bassi rispetto alle fasi successive del ciclo di vita. Questo approccio è in linea con il principio di sicurezza shift-left. In definitiva, la modellazione delle minacce si integra con il processo di gestione del rischio di un'organizzazione e aiuta a prendere decisioni sui controlli da implementare utilizzando un approccio orientato alle minacce.

Quando eseguire la modellazione delle minacce?

La modellazione delle minacce deve essere avviata il prima possibile nel ciclo di vita del carico di lavoro, in modo da avere una maggiore flessibilità di intervento sulle minacce identificate. Come per i bug del software, prima si identificano le minacce, più è conveniente affrontarle. Un modello di minacce è un documento vivo e deve continuare a evolvere in base ai cambiamenti dei carichi di lavoro. I modelli di minaccia vanno riesaminati nel tempo, anche in caso di modifiche importanti, di cambiamenti nel panorama delle minacce o di adozione di nuove funzionalità o servizi.

Passaggi dell'implementazione

In che modo è possibile eseguire la modellazione delle minacce?

Esistono diversi modi per eseguire la modellazione delle minacce. Come per i linguaggi di programmazione, anche in questo caso ci sono vantaggi e svantaggi e bisogna scegliere il metodo più adatto alle proprie esigenze. Un approccio consiste nell'iniziare con 4 domande per la modellazione delle minacce di Shostack, che pone domande aperte per fornire una struttura per il tuo esercizio di modellazione delle minacce:

1. A cosa si sta lavorando?

Questa domanda ha lo scopo di aiutare a comprendere e concordare il sistema che si sta costruendo e i dettagli di tale sistema che sono rilevanti per la sicurezza. La creazione di un modello o di un diagramma è la soluzione più comune per rispondere a questa domanda, in quanto consente di visualizzare ciò che si sta creando, ad esempio utilizzando un diagramma di flusso dei dati. Scrivere ipotesi e dettagli importanti del sistema aiuta anche a definire l'ambito di applicazione. In questo modo, tutti coloro che contribuiscono alla modellazione delle minacce possono concentrarsi sullo stesso aspetto, evitando deviazioni dispendiose in termini di tempo su argomenti fuori portata (comprese le versioni non aggiornate del sistema). Ad esempio, se si sta realizzando un'applicazione Web, probabilmente non vale la pena procedere alla modellazione per la sequenza di avvio attendibile del sistema operativo per i browser client, poiché non si ha la possibilità di influire su questo aspetto con il proprio progetto.

2. Che cosa può andare storto?

   In questa fase si identificano le minacce al sistema. Le minacce sono azioni o eventi accidentali o intenzionali che producono impatti indesiderati e potrebbero compromettere la sicurezza del sistema. Senza una visione chiara di ciò che potrebbe andare storto, non è possibile fare nulla per evitarlo.

   Non esiste un elenco canonico di ciò che può andare storto. La creazione di questo elenco richiede un brainstorming e la collaborazione tra tutte le persone del team e le persone pertinenti coinvolte nell'esercizio di modellazione delle minacce. Per semplificare il brainstorming, utilizza un modello per identificare le minacce, come STRIDE, che suggerisce diverse categorie da valutare: spoofing, manomissione, ripudio, divulgazione di informazioni, negazione del servizio ed elevazione dei privilegi. Inoltre, potresti contribuire al brainstorming consultando gli elenchi esistenti e traendone ispirazione, tra cui OWASP Top 10, HiTrust Threat Catalog e il catalogo delle minacce della tua organizzazione.

3. Cosa si intende fare al riguardo?

   Come nel caso della domanda precedente, non esiste un elenco canonico di tutte le possibili mitigazioni. Gli input di questa fase sono le minacce, gli attori e le aree di miglioramento identificate nella fase precedente.

   Sicurezza e conformità sono una responsabilità condivisa tra AWS e l'utente. È importante capire che quando si chiede "Che si farà al riguardo?", si chiede anche "Chi è responsabile? Chi ha la responsabilità di fare qualcosa?" Comprendere l'equilibrio delle responsabilità tra utente e AWS consente di limitare l'esercizio di modellazione delle minacce alle mitigazioni sotto il proprio

controllo, che di solito sono una combinazione di opzioni di configurazione del servizio AWS e di mitigazioni specifiche del proprio sistema.

In merito alla parte di responsabilità di AWS condivisa, scoprirai che i servizi AWS rientrano nell'ambito di molti programmi di conformità. Questi programmi aiutano a comprendere i solidi controlli in atto presso AWS per mantenere la sicurezza e la conformità del cloud. I report di audit di questi programmi possono essere scaricati per i clienti AWS da AWS Artifact.

Indipendentemente dai servizi AWS utilizzati, c'è sempre una responsabilità del cliente e le mitigazioni allineate a tale responsabilità devono essere incluse nel modello di minaccia. Per quanto riguarda le mitigazioni dei controlli di sicurezza per i servizi AWS stessi, è necessario considerare l'implementazione dei controlli di sicurezza in tutti i domini, compresi quelli quali la gestione delle identità e degli accessi (autenticazione e autorizzazione), la protezione dei dati (a riposo e in transito), la sicurezza dell'infrastruttura, la creazione di log e il monitoraggio. La documentazione di ciascun servizio AWS prevede un capitolo dedicato alla sicurezza, con indicazioni sui controlli di sicurezza da prendere in considerazione come mitigazioni. È importante considerare il codice che si sta scrivendo e le sue dipendenze e pensare ai controlli attuabili per affrontare queste minacce. Questi controlli potrebbero corrispondere a elementi quali la convalida degli input, la gestione delle sessioni e la gestione dei limiti. Spesso la maggior parte delle vulnerabilità viene introdotta nel codice personalizzato, quindi è bene concentrarsi su quest'area.

4. È stato fatto un buon lavoro?

   L'obiettivo è il miglioramento da parte del team e dell'organizzazione sia della qualità dei modelli di minacce sia della relativa velocità di esecuzione nel tempo. Questi miglioramenti derivano da una combinazione di pratica, apprendimento, insegnamento e revisione. Per approfondire e sperimentare nella pratica, è consigliabile che tu e il tuo team completiate il corso di formazione Threat modeling the right way for builders training course o il workshop. Inoltre, se stai cercando indicazioni su come integrare la modellazione delle minacce nel ciclo di vita di sviluppo delle applicazioni della tua organizzazione, consulta il post How to approach threat modeling sul blog di AWS sulla sicurezza.

Threat Composer

Come strumento di ausilio e guida nella modellazione delle minacce, prendi in considerazione l'utilizzo dello strumento Threat Composer, il cui scopo è ridurre il time-to-value di questa attività. Lo strumento consente di eseguire le seguenti operazioni:

- Scrivere dichiarazioni utili sulle minacce in linea con la sintassi delle minacce che funzionino in un flusso di lavoro naturale non lineare
- Generare un modello di minaccia leggibile dall'uomo
- Generare un modello di minaccia leggibile dal computer per consentire la gestione dei modelli di minaccia come codice
- Velocizzare l'individuazione delle aree di miglioramento della qualità e della copertura utilizzando l'area del pannello di controllo contenente le informazioni dettagliate

Per ulteriori informazioni, visita Threat Composer e passa all'area di lavoro esemplificativa definita dal sistema.

Risorse

Best practice correlate:

- SEC01-BP03 Identificazione e convalida degli obiettivi di controllo
- SEC01-BP04 Rimani aggiornato sulle minacce alla sicurezza e sui consigli
- SEC01-BP05 Ridurre l'ambito di gestione della sicurezza
- SEC01-BP08 Valutazione e implementazione periodiche di nuovi servizi e funzionalità di sicurezza

Documenti correlati:

- Come approcciare la modellazione delle minacce (AWS Security Blog)
- NIST: Guide to Data-Centric System Threat modeling

Video correlati:

- AWS Summit ANZ 2021 - How to approach threat modelling
- AWS Summit ANZ 2022 - Scaling security – Optimise for fast and secure delivery

Formazione correlata:

- Threat modeling the right way for builders – AWS Skill Builder virtual self-paced training
- Threat modeling the right way for builders – AWS Workshop

Strumenti correlati:

- Threat Composer

SEC01-BP08 Valutazione e implementazione periodiche di nuovi servizi e funzionalità di sicurezza

Valuta e implementa servizi e funzionalità di sicurezza di AWS e partner AWS che consentano di sviluppare l'assetto di sicurezza del carico di lavoro.

Risultato desiderato: hai adottato una procedura standard che ti informa su nuovi servizi e funzionalità rilasciati da AWS e dai partner AWS. Puoi valutare come queste nuove funzionalità influenzino la progettazione di controlli attuali e nuovi per i tuoi ambienti e carichi di lavoro.

Anti-pattern comuni:

- Non ti iscrivi ai blog e ai feed RSS di AWS per conoscere rapidamente le nuove funzionalità e i servizi più importanti.
- Fai affidamento su notizie e aggiornamenti sui servizi e sulle funzioni di sicurezza provenienti da fonti di seconda mano
- Non incoraggi gli utenti AWS della tua organizzazione a rimanere informati sugli ultimi aggiornamenti

Vantaggi dell'adozione di questa best practice: rimanere aggiornati sui nuovi servizi e funzionalità di sicurezza, consente di adottare decisioni informate sull'implementazione dei controlli negli ambienti cloud e nei carichi di lavoro. Queste origini contribuiscono ad aumentare la consapevolezza dell'evoluzione del panorama della sicurezza e di come i servizi AWS possano essere utilizzati per proteggersi dalle minacce nuove ed emergenti.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

AWS informa i clienti sui nuovi servizi e funzionalità di sicurezza attraverso diversi canali:

- AWS What's New
- AWS Blog delle novità
- AWS Security Blog
- AWS Security Bulletins

- AWS documentation overview

Puoi iscriverti a un argomento AWS Daily Feature Updates utilizzando Amazon Simple Notification Service (Amazon SNS) per un riepilogo giornaliero completo degli aggiornamenti. Alcuni servizi di sicurezza, come Amazon GuardDuty e AWS Security Hub CSPM, forniscono i propri argomenti SNS in modo da restare informati su nuovi standard, esiti e altri aggiornamenti di questi particolari servizi.

Anche durante conferenze, eventi e webinar che si tengono ogni anno in tutto il mondo, vengono annunciati nuovi servizi e funzionalità. Segnaliamo in particolare conferenza annuale sulla sicurezza AWS re:Inforce e la conferenza più generale AWS re:Invent. I canali di notizie AWS di cui sopra condividono questi annunci relativi a conferenze sulla sicurezza e su altri servizi. Inoltre, puoi guardare le sessioni breakout di approfondimento didattiche online sul canale AWS Events channel e su YouTube.

Puoi anche chiedere al team del tuo Account AWS informazioni sugli aggiornamenti e consigli più recenti sui servizi di sicurezza. Puoi contattare il team tramite il modulo Sales Support se non disponi dei loro recapiti diretti. Allo stesso modo, se sei abbonato al supporto AWS Enterprise, riceverai aggiornamenti settimanali dal tuo Technical Account Manager (TAM) e potrai programmare una riunione di revisione regolare con lo stesso.

Passaggi dell'implementazione

1. Iscriviti ai vari blog e bollettini con il tuo lettore RSS preferito o all'argomento Daily Features Updates SNS.
2. Valuta gli eventi AWS a cui partecipare per conoscere in prima persona nuove funzionalità e servizi.
3. Organizza riunioni con il team Account AWS per qualsiasi domanda sull'aggiornamento dei servizi e delle funzionalità di sicurezza.
4. Prendi in considerazione la possibilità di abbonarti a Enterprise Support per avere consulenze regolari con un Technical Account Manager (TAM).

Risorse

Best practice correlate:

- PERF01-BP01 Informazioni e identificazione dei servizi e delle funzionalità cloud disponibili
- COST01-BP07 Mantenimento dell'aggiornamento sulle nuove versioni dei servizi

Gestione dell’identità e degli accessi

Questions

- SEC 2. Come si gestisce l'autenticazione per persone e macchine?
- SEC 3. Come si gestiscono le autorizzazioni per persone e macchine?

SEC 2. Come si gestisce l'autenticazione per persone e macchine?

Ci sono due tipi di identità da gestire quando inizi a utilizzare carichi di lavoro AWS sicuri.

- Identità umane: le identità umane che richiedono l'accesso agli ambienti e alle applicazioni AWS possono essere classificate in tre gruppi, ossia forza lavoro, terze parti e utenti.

  Il gruppo della forza lavoro include amministratori, sviluppatori e operatori che sono membri dell'organizzazione. Hanno bisogno dell'accesso per gestire, creare e utilizzare le risorse AWS.

  Il gruppo delle terze parti include collaboratori esterni, come appaltatori, fornitori o partner. Interagiscono con le risorse AWS nell'ambito del loro rapporto di lavoro con l'organizzazione.

  Il gruppo degli utenti include i consumatori delle applicazioni. Accedono alle risorse AWS tramite browser web, applicazioni client, app per dispositivi mobili o strumenti da riga di comando interattivi.
- Identità di macchine: le applicazioni per il carico di lavoro, gli strumenti operativi e i componenti necessitano di un'identità per effettuare richieste ai servizi AWS, ad esempio la lettura dei dati. Queste identità includono anche macchine in esecuzione all'interno dell'ambiente AWS, come le istanze Amazon EC2 o le funzioni AWS Lambda. Puoi anche gestire le identità di macchine per soggetti esterni, o per macchine al di fuori di AWS, che richiedono l'accesso all'ambiente AWS.

Best practice

- SEC02-BP01 Utilizzo di meccanismi di accesso efficaci
- SEC02-BP02 Utilizzo di credenziali temporanee
- SEC02-BP03 Archiviazione e utilizzo dei segreti in modo sicuro
- SEC02-BP04 Fai affidamento su un gestore dell'identità digitale centralizzato
- SEC02-BP05 Verifica e rotazione periodica delle credenziali
- SEC02-BP06 Impiego dei gruppi di utenti e degli attributi

SEC02-BP01 Utilizzo di meccanismi di accesso efficaci

Gli accessi (autenticazione tramite credenziali di accesso) possono presentare dei rischi se non si utilizzano meccanismi come l'autenticazione a più fattori (MFA), soprattutto in situazioni in cui le credenziali di accesso sono state inavvertitamente divulgate o sono facilmente identificabili. Utilizza meccanismi di accesso efficaci per ridurre tali rischi, richiedendo l'MFA e policy sulle password sicure.

Risultato desiderato: ridurre i rischi di accessi accidentali alle credenziali AWS utilizzando meccanismi di accesso avanzati per gli utenti AWS Identity and Access Management (IAM), l'utente root Account AWS, AWS IAM Identity Center e i gestori dell'identità digitale di terze parti. Ciò significa richiedere l'MFA, applicare policy sulle password efficaci e rilevare comportamenti di accesso anomali.

Anti-pattern comuni:

- Nessuna applicazione di policy sulle password efficaci per le proprie identità, comprese password complesse e MFA.
- Condivisione delle stesse credenziali tra utenti diversi.
- Nessun utilizzo di controlli investigativi per gli accessi sospetti.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Ci sono diversi modi in cui le identità umane possono accedere a AWS. È una best practice di AWS affidarsi a un gestore dell'identità digitale centralizzato utilizzando la federazione (federazione diretta SAML 2.0 tra AWS IAM e l'IdP centralizzato o utilizzando Centro identità AWS IAM) per l'autenticazione ad AWS. In questo caso, stabilisci un processo di accesso sicuro con il gestore dell'identità digitale o con Microsoft Active Directory.

Quando apri un Account AWS, inizi con un utente root Account AWS. L'utente root dell'account va utilizzato solo per configurare l'accesso degli utenti (e per le attività che richiedono l'utente root). È importante attivare l'autenticazione a più fattori (MFA) per l'utente root dell'account subito dopo l'apertura dell'Account AWS e proteggere l'utente root utilizzando la guida alle best practice di AWS.

Centro identità AWS IAM è progettato per gli utenti della forza lavoro; puoi creare e gestire le identità degli utenti all'interno del servizio e proteggere il processo di accesso con l'MFA. AWS Cognito,

invece, è progettato per la gestione di identità e accessi del cliente (CIAM), che fornisce pool di utenti e gestore dell'identità digitale per le identità degli utenti esterni nelle applicazioni.

Se crei utenti in Centro identità AWS IAM, proteggi il processo di accesso in tale servizio e attiva MFA. Per le identità degli utenti esterni nelle applicazioni, puoi utilizzare i pool di utenti di Amazon Cognito e proteggere il processo di accesso in tale servizio o attraverso uno dei gestori dell'identità digitale supportati nei pool di utenti di Amazon Cognito.

Inoltre, per gli utenti in Centro identità AWS IAM, puoi utilizzare Accesso verificato da AWS per fornire un ulteriore livello di sicurezza, verificando l'identità e la postura del dispositivo dell'utente prima che venga concesso l'accesso alle risorse AWS.

Se utilizzi utenti AWS Identity and Access Management (IAM), proteggi il processo di accesso utilizzando IAM.

Puoi utilizzare contemporaneamente Centro identità AWS IAM e federazione diretta IAM per gestire l'accesso ad AWS. Puoi utilizzare la federazione IAM per gestire l'accesso a Console di gestione AWS e ai servizi e Centro identità IAM per gestire l'accesso ad applicazioni aziendali come QuickSight o Amazon Q Business.

Indipendentemente dal metodo di accesso, è fondamentale applicare una policy di accesso efficace.

Passaggi dell'implementazione

Di seguito sono indicate raccomandazioni generali per l'accesso sicuro. Configura le impostazioni effettive in base alla policy aziendale. In alternativa, utilizza uno standard, come NIST 800-63.

- Richiedi MFA. È una best practice IAM richiedere l'MFA per identità umane e carichi di lavoro. L'attivazione dell'MFA fornisce un ulteriore livello di sicurezza che richiede agli utenti di fornire le credenziali di accesso e un codice OTP (One-Time Password) o una stringa verificata e generata a livello crittografico da un dispositivo hardware.
- Applica una lunghezza minima della password, fattore primario nell'efficacia della password.
- Applica la complessità delle password in modo che sia più difficile individuarle.
- Consenti agli utenti di cambiare le loro password.
- Crea identità individuali invece di credenziali condivise. Creando identità individuali, puoi assegnare a ciascun utente un set unico di credenziali di sicurezza. I singoli utenti consentono di sottoporre ad audit l'attività di ciascuno.

Consigli del Centro identità IAM:

- Il Centro identità IAM offre una policy sulle password prestabilita in caso di utilizzo della directory predefinita che stabilisce lunghezza, complessità e requisiti di riutilizzo della password.
- Attiva l'MFA e configura l'impostazione relativa alla sensibilità al contesto o all'attivazione costante per l'MFA quando l'origine di identità è la directory predefinita, AWS Managed Microsoft AD o AD Connector.
- Consenti agli utenti di registrare i propri dispositivi MFA.

Consigli sulle directory dei pool di utenti Amazon Cognito:

- Configura le impostazioni relative alla complessità della password.
- Richiedi l'MFA per gli utenti.
- Le impostazioni di sicurezza avanzate dei pool di utenti di Amazon Cognito offrono funzionalità come l'autenticazione adattiva, che può bloccare gli accessi sospetti.

Suggerimenti per l'utente IAM:

- Idealmente stai utilizzando il Centro identità IAM o la federazione diretta. Tuttavia, potrebbero essere necessari utenti IAM. In tal caso, imposta una policy sulle password per gli utenti IAM. Puoi utilizzare la policy sulle password per definire requisiti quali la lunghezza minima o la necessità che la password richieda caratteri non alfabetici.
- Crea una policy IAM per applicare l'accesso MFA: in questo modo, gli utenti potranno gestire le proprie password e i propri dispositivi MFA.

Risorse

Best practice correlate:

- SEC02-BP03 Archiviazione e utilizzo dei segreti in modo sicuro
- SEC02-BP04 Fai affidamento su un gestore dell'identità digitale centralizzato
- SEC03-BP08 Condivisione delle risorse in modo sicuro all'interno dell'organizzazione

Documenti correlati:

- AWS IAM Identity Center Password Policy
- IAM user password policy

- Setting the Account AWS root user password
- Amazon Cognito password policy
- AWS credentials
- IAM security best practices

Video correlati:

- Managing user permissions at scale with AWS IAM Identity Center
- Mastering identity at every layer of the cake

SEC02-BP02 Utilizzo di credenziali temporanee

Quando si esegue qualsiasi tipo di autenticazione, è preferibile utilizzare credenziali temporanee invece di credenziali a lungo termine per ridurre o eliminare i rischi, come la divulgazione, la condivisione o il furto involontario delle stesse.

Risultato desiderato: al fine di ridurre il rischio di credenziali a lungo termine, utilizza credenziali temporanee laddove possibile per le identità di persone e macchine. Le credenziali a lungo termine creano molti rischi, come l'esposizione attraverso i caricamenti su repository pubblici. Grazie alle credenziali temporanee, riduci notevolmente le possibilità di compromissione delle credenziali.

Anti-pattern comuni:

- Sviluppatori che utilizzano chiavi di accesso a lungo termine dagli utenti IAM anziché ottenere credenziali temporanee dalla CLI utilizzando la federazione.
- Sviluppatori che inseriscono chiavi di accesso a lungo termine nel loro codice e caricano tale codice su repository Git pubblici.
- Sviluppatori che inseriscono chiavi di accesso a lungo termine nelle app mobili che vengono poi rese disponibili negli app store.
- Utenti che condividono le chiavi di accesso a lungo termine con altri utenti o dipendenti che lasciano l'azienda con chiavi di accesso a lungo termine ancora in loro possesso.
- Utilizzo di chiavi di accesso a lungo termine per le identità macchina quando è possibile utilizzare credenziali temporanee.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Utilizza credenziali di sicurezza temporanee anziché credenziali a lungo termine per tutte le richieste API e CLI AWS. In quasi tutti i casi, le richieste API e CLI rivolte ai servizi AWS devono essere firmate mediante chiavi di accesso AWS. Queste richieste possono essere firmate con credenziali temporanee o a lungo termine. L'unico caso in cui occorre utilizzare credenziali a lungo termine, note anche come chiavi di accesso a lungo termine, è l'utilizzo di utenti IAM o dell'utente root Account AWS. L'utilizzo della federazione per AWS o l'assunzione di un ruolo IAM tramite altri metodi prevede la creazione di credenziali temporanee. Anche quando accedi a Console di gestione AWS utilizzando le credenziali di accesso, vengono generate credenziali temporanee per effettuare chiamate ai servizi AWS. Sono poche le situazioni in cui occorrono credenziali a lungo termine ed è possibile svolgere quasi tutte le attività utilizzando credenziali temporanee.

Evitare l'uso di credenziali a lungo termine a favore di credenziali temporanee dovrebbe andare di pari passo con una strategia di riduzione dell'uso degli utenti IAM a favore della federazione e dei ruoli IAM. Sebbene l'utilizzo in passato degli utenti IAM sia per le identità umane che per quelle macchina, ora si consiglia di non utilizzarli per evitare i rischi legati all'uso di chiavi di accesso a lungo termine.

Passaggi dell'implementazione

Identità umane

Per le identità della forza lavoro come dipendenti, amministratori, sviluppatori e operatori:

- Dovresti affidarti a gestori dell'identità digitale centralizzati e richiedere agli utenti umani di utilizzare la federazione con un gestore dell'identità digitale per accedere ad AWS utilizzando credenziali temporanee. La federazione per gli utenti può essere effettuata sia con la federazione diretta a ciascun Account AWS sia utilizzando Centro identità AWS IAM e il gestore dell'identità digitale preferito. La federazione offre una serie di vantaggi rispetto all'utilizzo degli utenti IAM, oltre all'eliminazione delle credenziali a lungo termine. I tuoi utenti possono inoltre richiedere credenziali temporanee dalla riga di comando per la federazione diretta o utilizzando il Centro identità IAM. Ciò significa che i casi d'uso che richiedono utenti IAM o credenziali a lungo termine per gli utenti sono pochi.

Per le identità di terze parti:

- Quando concedi l'accesso alle risorse del tuo Account AWS a terze parti, come i fornitori di software as a service (SaaS), puoi utilizzare ruoli multi-account e policy basate su risorse. Inoltre,

puoi utilizzare il flusso delle credenziali client di concessione di Amazon Cognito OAuth 2.0 per clienti o partner SaaS B2B.

Identità utente che accedono alle risorse AWS tramite browser web, applicazioni client, app per dispositivi mobili o strumenti interattivi da riga di comando:

- Se devi concedere alle applicazioni per consumatori o clienti l'accesso alle tue risorse AWS, puoi utilizzare i pool di identità di Amazon Cognito o i pool di utenti Amazon Cognito per fornire credenziali temporanee. Le autorizzazioni per le credenziali sono configurate attraverso i ruoli IAM. Puoi inoltre definire un ruolo IAM separato con autorizzazioni limitate per gli utenti guest non autenticati.

Identità macchina

Per le identità macchina, potrebbero essere necessarie credenziali a lungo termine. In questi casi, dovresti richiedere ai carichi di lavoro di utilizzare credenziali temporanee con ruoli IAM per l'accesso ad AWS.

- Per Amazon Elastic Compute Cloud (Amazon EC2), puoi utilizzare i ruoli per Amazon EC2.
- AWS Lambda ti consente di configurare un ruolo di esecuzione Lambda per la concessione delle autorizzazioni di servizio al fine di eseguire azioni AWS mediante credenziali temporanee. Per i servizi AWS esistono molti altri modelli simili per concedere credenziali temporanee utilizzando i ruoli IAM.
- Per i dispositivi IoT, puoi richiedere credenziali temporanee al provider di credenziali AWS IoT Core.
- Per sistemi on-premises o quelli eseguiti all'esterno di AWS che richiedono l'accesso alle risorse AWS, puoi utilizzare IAM Roles Anywhere.

Esistono scenari in cui le credenziali temporanee non sono supportate e che richiedono l'uso di credenziali a lungo termine. In queste situazioni, verifica e ruota periodicamente queste credenziali e ruota regolarmente le chiavi di accesso. Per chiavi di accesso dell'utente IAM altamente limitate, considera le seguenti misure di sicurezza aggiuntive:

- Concedi autorizzazioni altamente limitate:
  - Rispetta il principio del privilegio minimo (con impostazioni specifiche per azioni, risorse e condizioni).

  - Valuta la possibilità di concedere all'utente IAM solo l'operazione AssumeRole per un ruolo specifico. A seconda dell'architettura on-premises, questo approccio consente di isolare e proteggere le credenziali IAM a lungo termine.
- Limita le origini della rete e gli indirizzi IP consentiti nella policy di attendibilità dei ruoli IAM.
- Monitora l'utilizzo e imposta avvisi per le autorizzazioni non utilizzate o l'uso improprio (utilizzando i filtri metriche e gli allarmi di AWS CloudWatch Logs).
- Applica i limiti delle autorizzazioni (le policy di controllo dei servizi (SCP) e i limiti delle autorizzazioni si completano a vicenda: le SCP sono poco granulari, mentre i limiti delle autorizzazioni sono più granulari).
- Implementa un processo per il provisioning e l'archiviazione sicura (in vault on-premises) delle credenziali.

Altre opzioni per gli scenari che richiedono credenziali a lungo termine sono le seguenti:

- Crea la tua API di distribuzione di token (utilizzando Gateway Amazon API).
- Per gli scenari in cui è necessario utilizzare credenziali a lungo termine o credenziali diverse dalle chiavi di accesso AWS (come i login ai database), puoi utilizzare un servizio progettato per gestire i segreti, come Gestione dei segreti AWS. Secrets Manager semplifica la gestione, la rotazione e l'archiviazione sicura dei segreti crittografati. Molti servizi AWS supportano l'integrazione diretta con Secrets Manager.
- Per le integrazioni multi-cloud, puoi utilizzare la federazione delle identità basata sulle credenziali del provider di servizi di credenziali (CSP) di origine (consulta AWS STS AssumeRoleWithWebIdentity).

Per ulteriori informazioni sulla rotazione delle credenziali a lungo termine, consulta rotazione delle chiavi di accesso.

Risorse

Best practice correlate:

- SEC02-BP03 Archiviazione e utilizzo dei segreti in modo sicuro
- SEC02-BP04 Fai affidamento su un gestore dell'identità digitale centralizzato
- SEC03-BP08 Condivisione delle risorse in modo sicuro all'interno dell'organizzazione

Documenti correlati:

- Temporary Security Credentials
- AWS Credenziali
- IAM Security Best Practices
- Ruoli IAM
- Centro identità IAM
- Identity Providers and Federation
- Rotating Access Keys
- Soluzioni dei partner per la sicurezza: accesso e controllo degli accessi
- L'utente root dell'account AWS
- Access AWS using a Google Cloud Platform native workload identity
- How to access AWS resources from Microsoft Entra ID tenants using AWS Security Token Service

Video correlati:

- Managing user permissions at scale with AWS IAM Identity Center
- Mastering identity at every layer of the cake

SEC02-BP03 Archiviazione e utilizzo dei segreti in modo sicuro

Un carico di lavoro richiede una capacità automatizzata di dimostrare la propria identità a database, risorse e servizi di terze parti. A tal fine, si utilizzano credenziali di accesso segrete, come chiavi di accesso API, password e token OAuth. L'utilizzo di un servizio appositamente creato per archiviare, gestire e ruotare queste credenziali aiuta a ridurre la probabilità che queste vengano compromesse.

Risultato desiderato: implementazione di un meccanismo per la gestione sicura delle credenziali delle applicazioni che consenta di raggiungere i seguenti obiettivi.

- Identificare i segreti necessari per il carico di lavoro.
- Ridurre il numero di credenziali a lungo termine sostituendole con credenziali a breve termine, laddove possibile.
- Stabilire l'archiviazione sicura e la rotazione automatica delle rimanenti credenziali a lungo termine.
- Sottoporre a audit l'accesso ai segreti esistenti nel carico di lavoro.

- Eseguire il monitoraggio continuo per verificare che nessun segreto sia incorporato nel codice sorgente durante il processo di sviluppo.
- Ridurre la probabilità che le credenziali vengano divulgate inavvertitamente.

Anti-pattern comuni:

- Nessuna rotazione delle credenziali.
- Memorizzazione di credenziali a lungo termine nel codice sorgente o nei file di configurazione.
- Memorizzazione delle credenziali a riposo non criptate.

Vantaggi dell'adozione di questa best practice:

- I segreti sono conservati in modo criptato a riposo e in transito.
- L'accesso alle credenziali è controllato tramite un'API (immaginala come un distributore automatico di credenziali).
- L'accesso alle credenziali (sia in lettura che in scrittura) viene sottoposto a audit e registrato.
- Separazione delle preoccupazioni: la rotazione delle credenziali viene eseguita da un componente distinto, che può essere segregato dal resto dell'architettura.
- La distribuzione dei segreti avviene in automatico on demand ai componenti software e la rotazione avviene in una posizione centrale.
- È possibile controllare l'accesso alle credenziali in modo granulare.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

In passato, le credenziali utilizzate per l'autenticazione ai database, alle API di terze parti, ai token e ad altri segreti potevano essere incorporate nel codice sorgente o nei file di ambiente. AWS fornisce diversi meccanismi per memorizzare queste credenziali in modo sicuro, ruotarle in automatico e sottoporre a audit il loro utilizzo.

Il modo migliore per affrontare la gestione dei segreti è seguire le indicazioni relative a rimozione, sostituzione e rotazione. Le credenziali più sicure sono quelle che non si devono memorizzare, gestire o trattare. Possono esserci credenziali non più necessarie per il funzionamento del carico di lavoro e che possono essere rimosse in modo sicuro.

Per le credenziali ancora necessarie per il corretto funzionamento del carico di lavoro, potrebbe esserci l'opportunità di sostituire le credenziali a lungo termine con credenziali temporanee o a breve termine. Ad esempio, invece di una codifica fissa di una chiave di accesso segreta AWS, si può pensare di sostituire le credenziali a lungo termine con credenziali temporanee utilizzando i ruoli IAM.

Alcuni segreti di lunga durata potrebbero non poter essere rimossi o sostituiti. È possibile archiviare tali segreti in un servizio come Gestione dei segreti AWS, dove saranno archiviati, gestiti e rotati a livello centrale su base regolare.

Un audit del codice sorgente e dei file di configurazione del carico di lavoro può rivelare molti tipi di credenziali. La tabella seguente riassume le strategie per gestire i tipi più comuni di credenziali:

Tipo di credenziali	Descrizione	Strategia suggerita
Chiavi di accesso IAM	Chiavi segrete e accesso IAM AWS utilizzate per assumere ruoli IAM all'interno di un carico di lavoro	Sostituzione: utilizza invece i ruoli IAM assegnati alle istanze di calcolo (come Amazon EC2 o AWS Lambda). Per l'interoperabilità con terze parti che richiedon o l'accesso alle risorse del tuo Account AWS, chiedi se supportano l'accesso multi-account AWS. Per le app mobili, prendi in considera zione l'utilizzo di credenziali temporanee tramite pool di identità di Amazon Cognito (identità federate). Per i carichi di lavoro eseguiti all'esterno di AWS, valuta IAM Roles Anywhere o le attivazio ni ibride di AWS Systems Manager. Per i container, consulta il ruolo IAM dell'atti vità di Amazon ECS o il ruolo IAM del nodo di Amazon ECS.

Tipo di credenziali	Descrizione	Strategia suggerita
Chiavi SSH	Chiavi private Secure Shell utilizzate per accedere alle istanze Linux EC2, manualmente o nell'ambito di un processo automatizzato	Sostituzione: utilizza AWS Systems Manager o EC2 Instance Connect per fornire un accesso programmatico e umano alle istanze EC2 mediante i ruoli IAM.
Credenziali di applicazione e database	Password: stringa di testo semplice	Rotazione: memorizza le credenziali in Gestione dei segreti AWS e, laddove possibile, stabilisci una rotazione automatica.
Credenziali del database di amministrazione Aurora e Amazon RDS	Password: stringa di testo semplice	Sostituzione: utilizza l'integrazione di Secrets Manager con Amazon RDS o Amazon Aurora. Inoltre, alcuni tipi di database RDS possono utilizzare i ruoli IAM anziché le password per alcuni casi d'uso (per maggiori dettagli, consulta Autenticazione del database IAM).
Token OAuth	Token segreti: stringa di testo semplice	Rotazione: archivia i token in Gestione dei segreti AWS e configura la rotazione automatica.
Token e chiavi API	Token segreti: stringa di testo semplice	Rotazione: archivia in Gestione dei segreti AWS e stabilisci una rotazione automatica, laddove possibile.

Un anti-pattern comune è quello di incorporare le chiavi di accesso IAM all'interno del codice sorgente, dei file di configurazione o delle applicazioni mobili. Se occorre una chiave di accesso IAM per la comunicazione con un servizio AWS, utilizza credenziali di sicurezza temporanee (a breve termine). È possibile fornire queste credenziali a breve termine tramite ruoli IAM per le istanze EC2, ruoli di esecuzione per le funzioni Lambda, ruoli IAM di Cognito per l'accesso degli utenti mobili e policy IoT Core per i dispositivi IoT. Nell'interfacciarsi con terze parti, è preferibile delegare l'accesso a un ruolo IAM con l'accesso necessario alle risorse dell'account anziché configurare un utente IAM e inviare alla terza parte la chiave di accesso segreta per l'utente interessato.

Esistono molti casi in cui il carico di lavoro richiede l'archiviazione dei segreti necessari per l'interoperabilità con altri servizi e risorse. Gestione dei segreti AWS è stato creato proprio per gestire in modo sicuro queste credenziali, nonché l'archiviazione, l'uso e la rotazione di token API, password e altre credenziali.

Gestione dei segreti AWS offre cinque funzionalità chiave per garantire la sicurezza di archiviazione e gestione delle credenziali sensibili: crittografia a riposo, crittografia in transito, audit completi, controllo granulare degli accessi e rotazione delle credenziali estensibile. Sono accettabili anche altri servizi di gestione dei segreti dei partner AWS o soluzioni sviluppate localmente che forniscano funzionalità e garanzie simili.

Quando si recupera un segreto, è possibile utilizzare il componente di caching lato client di Secrets Manager per memorizzarlo nella cache per un uso futuro. Il recupero di un segreto memorizzato nella cache è più veloce rispetto al recupero da Secrets Manager. Inoltre, poiché la chiamata alle API di Secrets Manager ha un costo, l'uso della cache può ridurre i costi. Per una descrizione di tutti i modi in cui è possibile recuperare i segreti, consulta Ottieni segreti.

> **Note**
>
> Alcune lingue possono richiedere l'implementazione di una propria crittografia in memoria per la cache lato client.

Passaggi dell'implementazione

1. Identifica i percorsi di codice con credenziali con codifica fissa mediante strumenti automatizzati come Amazon CodeGuru.
   a. Utilizza Amazon CodeGuru per eseguire la scansione dei repository di codice. Una volta completata l'analisi, filtra su Type=Secrets in CodeGuru per trovare righe di codice problematiche.

2. Identifica le credenziali che possono essere rimosse o sostituite.
   a. Identifica le credenziali non più necessarie e contrassegnarle per la rimozione.
   b. Le chiavi segrete AWS incorporate nel codice sorgente devono essere sostituite con ruoli IAM associati alle risorse necessarie. Se parte del tuo carico di lavoro è al di fuori di AWS ma richiede credenziali IAM per accedere a risorse AWS, prendi in considerazione IAM Roles Anywhere o le attivazioni ibride di AWS Systems Manager.
3. Per altri segreti di terze parti a lunga durata che richiedono l'uso della strategia di rotazione, integra Secrets Manager nel codice per recuperare i segreti di terze parti in fase di esecuzione.
   a. La console di CodeGuru può creare in automatico un segreto in Secrets Manager utilizzando le credenziali scoperte.
   b. Integra il recupero dei segreti da Secrets Manager nel codice dell'applicazione.
      i. Le funzioni Lambda serverless possono utilizzare un'estensione Lambda indipendente dal linguaggio.
      ii. Per container o istanze EC2, AWS fornisce un esempio di codice lato client per il recupero di segreti da Secrets Manager in diversi linguaggi di programmazione diffusi.
4. Esamina periodicamente la base di codice e ripetere la scansione per verificare che non siano stati aggiunti nuovi segreti al codice.
   a. Prendi in considerazione l'utilizzo di uno strumento come git-secrets per evitare di inserire nuovi segreti nel tuo repository di codice sorgente.
5. Monitora l'attività di Secrets Manager per individuare eventuali indicazioni di utilizzo imprevisto, accesso inopportuno ai segreti o tentativi di eliminazione degli stessi.
6. Riduci l'esposizione umana alle credenziali. Limita l'accesso alle credenziali di lettura, scrittura e modifica a un ruolo IAM dedicato a questo scopo e fornisci l'accesso in modo che il ruolo sia assunto solo da un piccolo sottoinsieme di utenti operativi.

Risorse

Best practice correlate:

- SEC02-BP02 Utilizzo di credenziali temporanee
- SEC02-BP05 Verifica e rotazione periodica delle credenziali

Documenti correlati:

- Nozioni di base su Gestione dei segreti AWS

- Identity Providers and Federation
- Amazon CodeGuru Introduces Secrets Detector
- How Gestione dei segreti AWS uses AWS Key Management Service
- Secret encryption and decryption in Secrets Manager
- Articoli del blog su Secrets Manager
- Amazon RDS announces integration with Gestione dei segreti AWS

Video correlati:

- Best Practices for Managing, Retrieving, and Rotating Secrets at Scale
- Find Hard-Coded Secrets Using Amazon CodeGuru Secrets Detector
- Securing Secrets for Hybrid Workloads Using Gestione dei segreti AWS

Workshop correlati:

- Store, retrieve, and manage sensitive credentials in Gestione dei segreti AWS
- AWS Systems Manager Hybrid Activations

SEC02-BP04 Fai affidamento su un gestore dell'identità digitale centralizzato

Per le identità della forza lavoro (dipendenti e collaboratori) affidati a un gestore dell'identità digitale che ti consenta di gestire le identità in un luogo centralizzato. In questo modo è più semplice gestire l'accesso tra più applicazioni e sistemi, poiché crei, assegni, gestisci, revochi e verifichi gli accessi da una singola posizione.

Risultato desiderato: hai un gestore dell'identità digitale dal quale gestisci centralmente gli utenti della forza lavoro, le policy di autenticazione (come le richieste di autenticazione a più fattori (MFA)) e le autorizzazioni per sistemi e applicazioni, come l'assegnazione dell'accesso in base all'appartenenza o agli attributi di un utente. Gli utenti che fanno parte della tua forza lavoro accedono al gestore dell'identità digitale centrale ed effettuano l'accesso federato (autenticazione unica) alle applicazioni interne ed esterne, il che elimina la necessità per gli utenti di ricordare più credenziali. Il gestore dell'identità digitale è integrato con i tuoi sistemi di risorse umane (HR), in modo che le modifiche relative al personale vengano sincronizzate in automatico con il gestore dell'identità digitale. Ad esempio, se qualcuno lascia l'organizzazione, puoi revocare automaticamente l'accesso alle applicazioni e ai sistemi federati (incluso AWS). Hai abilitato la verifica dettagliata dei log nel tuo

gestore dell'identità digitale e stai monitorando questi log per rilevare comportamenti degli utenti insoliti.

Anti-pattern comuni:

- Non utilizzi federazione e autenticazione unica. Gli utenti che appartengono alla tua forza lavoro creano account utente e credenziali separati in più applicazioni e sistemi.
- Non hai automatizzato il ciclo di vita delle identità degli utenti che fanno parte della tua forza lavoro, ad esempio integrando il gestore dell'identità digitale con i tuoi sistemi HR. Quando un utente lascia l'organizzazione o cambia ruolo, segui una procedura manuale per eliminare o aggiornare i suoi record in più applicazioni e sistemi.

Vantaggi dell'adozione di questa best practice: utilizzare un gestore dell'identità digitale centralizzato ti fornisce un'unica piattaforma per gestire le identità e le policy degli utenti che fanno parte della tua forza lavoro, la possibilità di assegnare l'accesso alle applicazioni a utenti e gruppi e di monitorare l'attività di accesso degli utenti. Grazie all'integrazione con i sistemi di risorse umane (HR), quando un utente cambia ruolo, queste modifiche vengono sincronizzate con il gestore dell'identità digitale e le applicazioni e le autorizzazioni assegnate si aggiornano in automatico. Quando un utente lascia l'organizzazione, la sua identità viene automaticamente disabilitata nel gestore dell'identità digitale e l'accesso alle applicazioni e ai sistemi federati viene revocato.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Guida per gli utenti della forza lavoro che accedono a AWS Gli utenti della forza lavoro, come i dipendenti e i collaboratori dell'organizzazione, possono richiedere l'accesso a AWS utilizzando la Console di gestione AWS o AWS Command Line Interface (AWS CLI) per svolgere le mansioni lavorative. Puoi concedere l'accesso ad AWS a tali utenti federando il tuo gestore dell'identità digitale centralizzato AWS a due livelli: federazione diretta a ciascun Account AWS o federazione a più account della tua organizzazione AWS.

Per federare gli utenti della tua forza lavoro direttamente con ciascun Account AWS, utilizza un gestore dell'identità digitale centralizzato per federare l'accesso ad AWS Identity and Access Management in tale account. Grazie alla sua flessibilità, IAM ti consente di abilitare un gestore dell'identità digitale SAML 2.0 o Open ID Connect (OIDC) separato per ciascun Account AWS e di utilizzare attributi per gli utenti federati al fine di controllare gli accessi. Gli utenti della tua forza lavoro utilizzano il proprio browser Web per accedere al gestore dell'identità digitale e forniscono

le proprie credenziali (come password e codici token MFA). Il gestore dell'identità digitale rilascia un'asserzione SAML nel browser che viene inviata all'URL di accesso della Console di gestione AWS, così da consentire all'utente di accedere mediante l'autenticazione unica alla Console di gestione AWS assumendo un ruolo IAM. I tuoi utenti possono anche ottenere credenziali API AWS temporanee da AWS CLI o AWS SDK di AWS STS assumendo il ruolo IAM mediante un'asserzione SAML proveniente dal gestore dell'identità digitale.

Per federare gli utenti della forza lavoro con più account all'interno dell'organizzazione AWS, puoi usare Centro identità AWS IAM per gestire a livello centrale l'accesso degli utenti della forza lavoro agli Account AWS e alle applicazioni. Puoi abilitare il Centro identità per la tua organizzazione e configurare la tua origine di identità. Centro identità IAM fornisce una directory di origine di identità predefinita, utilizzabile per gestire utenti e gruppi. In alternativa, puoi scegliere un'origine di identità esterna connettendoti al tuo gestore dell'identità digitale esterno tramite SAML 2.0 e allocando in automatico utenti e gruppi tramite SCIM oppure connettendoti alla tua directory Microsoft AD mediante Directory Service. Una volta configurata un'origine di identità, puoi assegnare l'accesso agli Account AWS a utenti e gruppi, definendo policy di privilegio minimo nei tuoi set di autorizzazioni. Gli utenti della tua forza lavoro possono autenticarsi tramite il tuo gestore dell'identità digitale centrale per accedere al portale di accesso AWS ed eseguire l'accesso tramite autenticazione unica per gli Account AWS e le applicazioni cloud a loro assegnate. Gli utenti possono configurare AWS CLI v2 per l'autenticazione con il Centro identità e ottenere le credenziali per eseguire i comandi AWS CLI. Il Centro identità consente inoltre l'accesso tramite SSO ad applicazioni AWS, come Amazon SageMaker Studio IA e i portali Sitewise AWS IoT.

Dopo aver seguito le indicazioni precedenti, gli utenti della forza lavoro non avranno più bisogno di utilizzare utenti IAM e gruppi per le normali operazioni quando gestiscono i carichi di lavoro su AWS. Gli utenti e i gruppi sono infatti gestiti all'esterno di AWS e sono in grado di accedere alle risorse AWS come identità federata. Le identità federate utilizzano i gruppi definiti dal gestore dell'identità digitale centralizzato. Devi identificare e rimuovere gruppi IAM, utenti IAM e credenziali utente di lunga durata (password e chiavi di accesso) non più necessarie nei tuoi Account AWS. Puoi trovare le credenziali inutilizzate mediante i report sulle credenziali IAM, eliminare gli utenti IAM corrispondenti ed eliminare i gruppi IAM. Puoi applicare una policy di controllo dei servizi alla tua organizzazione, così da prevenire la creazione di nuovi gruppi e utenti IAM, imponendo che l'accesso ad AWS avvenga tramite identità federate.

> **Note**
>
> L'utente è responsabile della gestione della rotazione dei token di accesso SCIM, come descritto nella documentazione sul provisioning automatico. Inoltre, l'utente è responsabile della rotazione dei certificati a supporto della federazione delle identità.

Guida per gli utenti delle applicazioni Puoi gestire le identità degli utenti delle applicazioni, ad esempio di un'applicazione per dispositivi mobili, utilizzando Amazon Cognito come gestore dell'identità digitale centralizzato. Amazon Cognito consente l'autenticazione, autorizzazione e gestione degli utenti per le app Web e per dispositivi mobili. Amazon Cognito offre un archivio di identità scalabile fino a milioni di utenti, supporta la federazione delle identità sociali e aziendali e offre funzionalità di sicurezza avanzate per proteggere i tuoi utenti e la tua azienda. Puoi integrare la tua applicazione Web o mobile personalizzata con Amazon Cognito per aggiungere l'autenticazione degli utenti e il controllo degli accessi alle applicazioni in pochi minuti. Amazon Cognito si fonda su standard di identità aperti come SAML e Open ID Connect (OIDC), supporta varie normative di conformità e si integra con le risorse di sviluppo frontend e backend.

Passaggi dell'implementazione

Passaggi per l'accesso ad AWS degli utenti della forza lavoro

- Federa l'accesso ad AWS degli utenti della tua forza lavoro tramite un gestore dell'identità digitale centralizzato seguendo uno dei seguenti approcci:
  - Utilizza il Centro identità IAM per abilitare l'accesso tramite autenticazione unica in più Account AWS nella tua organizzazione AWS con la federazione del tuo gestore dell'identità digitale.
  - Utilizza IAM per connettere il gestore dell'identità digitale direttamente a ciascun Account AWS, così da consentire un accesso federato e granulare.
- Identifica e rimuovi gruppi e utenti IAM sostituiti da identità federate.

Passaggi per gli utenti delle tue applicazioni

- Utilizza Amazon Cognito come gestore dell'identità digitale centralizzato per le tue applicazioni.
- Integra le tue applicazioni personalizzate con Amazon Cognito mediante OpenID Connect e OAuth. Puoi sviluppare applicazioni personalizzate utilizzando le librerie Amplify, che forniscono interfacce semplici da integrare con una varietà di servizi AWS per l'autenticazione, come Amazon Cognito.

Risorse

Best practice correlate:

- SEC02-BP06 Impiego dei gruppi di utenti e degli attributi
- SEC03-BP02 Concessione dell'accesso con privilegio minimo
- SEC03-BP06 Gestione degli accessi in base al ciclo di vita

Documenti correlati:

- Federazione delle identità in AWS
- Best practice per la sicurezza in IAM
- Best practice AWS Identity and Access Management
- Getting started with IAM Identity Center delegated administration
- How to use customer managed policies in IAM Identity Center for advanced use cases
- AWS CLI v2: provider di credenziali del Centro identità IAM

Video correlati:

- AWS re:Inforce 2022 - AWS Identity and Access Management (IAM) deep dive
- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center
- AWS re:Invent 2018: Mastering Identity at Every Layer of the Cake

Esempi correlati:

- Workshop: Using AWS IAM Identity Center to achieve strong identity management

Strumenti correlati:

- AWS Security Competency Partner con competenze nella sicurezza: gestione di identità e accessi
- saml2aws

SEC02-BP05 Verifica e rotazione periodica delle credenziali

Sottoponi a audit e ruota periodicamente le credenziali per limitarne il tempo di utilizzo per l'accesso alle risorse. Le credenziali a lungo termine espongono a molti rischi, riducibili mediante la rotazione periodica.

Risultato desiderato: implementa la rotazione delle credenziali per ridurre i rischi associati all'utilizzo delle credenziali a lungo termine. Esegui regolarmente l'audit e rimedia alla non conformità con le policy di rotazione delle credenziali.

Anti-pattern comuni:

- Nessun audit dell'uso delle credenziali.
- Utilizzo non necessario di credenziali a lungo termine.
- Utilizzo di credenziali a lungo termine e mancata rotazione regolare.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Quando non è possibile fare affidamento sulle credenziali temporanee e occorrono credenziali a lungo termine, esegui l'audit delle credenziali per garantire l'applicazione dei controlli prestabiliti, ad esempio l'autenticazione a più fattori (MFA), la regolare rotazione e un livello di accesso appropriato.

La convalida periodica, preferibilmente tramite uno strumento automatizzato, è necessaria per verificare l'applicazione dei controlli corretti. Per le identità umane, è necessario richiedere agli utenti di modificare periodicamente le password e ritirare le chiavi di accesso a favore delle credenziali temporanee. Nel passaggio dagli utenti AWS Identity and Access Management (IAM) alle identità centralizzate, puoi creare report sulle credenziali per controllare gli utenti.

Ti consigliamo inoltre di monitorare l'MFA nel tuo gestore dell'identità digitale. È possibile configurare Regole di AWS Config o utilizzare gli standard di sicurezza AWS Security Hub CSPM per monitorare se gli utenti hanno configurato l'MFA. Valuta la possibilità di utilizzare IAM Roles Anywhere per fornire credenziali temporanee per le identità macchina. Nelle situazioni in cui l'utilizzo di credenziali temporanee e ruoli IAM non è possibile, sono necessari audit e rotazione frequenti delle chiavi di accesso.

Passaggi dell'implementazione

- Esegui con regolarità audit delle credenziali: l'audit delle identità configurate nel tuo gestore dell'identità digitale e in IAM consente di verificare che l'accesso al tuo carico di lavoro sia concesso solo alle identità autorizzate. Tali identità possono includere, a titolo esemplificativo ma non esaustivo, utenti IAM, utenti del Centro identità AWS IAM, utenti di Active Directory o utenti di altri gestori dell'identità digitale upstream. Ad esempio, eliminare le persone che lasciano l'organizzazione e i ruoli multi-account non più necessari. Predisponi un processo per sottoporre periodicamente ad audit le autorizzazioni ai servizi a cui accede un'entità IAM. In questo modo potrai identificare le policy da modificare per rimuovere le autorizzazioni non utilizzate. Utilizza i report delle credenziali e AWS Identity and Access Management Access Analyzer per eseguire l'audit di credenziali e autorizzazioni IAM. Usa Amazon CloudWatch per configurare allarmi per chiamate API specifiche chiamate all'interno del tuo ambiente AWS. Amazon GuardDuty può inoltre avvisarti in caso attività impreviste, possibili segnali di un accesso eccessivamente permissivo o un accesso non intenzionale alle credenziali IAM.
- Ruota le credenziali regolarmente: se non puoi utilizzare credenziali temporanee, ruota con regolarità le chiavi di accesso IAM a lungo termine (massimo ogni 90 giorni). In caso di divulgazione involontaria e a propria insaputa di una chiave di accesso, questo limita la durata di utilizzo delle credenziali per accedere alle risorse. Per informazioni sulla rotazione delle chiavi di accesso per gli utenti IAM, consulta Rotazione delle chiavi di accesso.
- Rivedi le autorizzazioni IAM: per migliorare la sicurezza del tuo Account AWS, rivedi con regolarità e monitora ciascuna policy IAM. Verifica che le policy rispettino il principio del privilegio minimo.
- Valuta la possibilità di automatizzare la creazione e gli aggiornamenti delle risorse IAM: il Centro identità IAM automatizza molte attività IAM, come la gestione di ruoli e policy. In alternativa, AWS CloudFormation può essere utilizzato per automatizzare l'implementazione delle risorse IAM, compresi ruoli e policy, per ridurre la possibilità di errore umano, poiché i modelli possono essere verificati e controllati in versione.
- Usa IAM Roles Anywhere per sostituire gli utenti IAM per le identità macchina: IAM Roles Anywhere consente di utilizzare i ruoli in aree in cui prima non era possibile, come i server on-premises. IAM Roles Anywhere utilizza un certificato X.509 attendibile per l'autenticazione a AWS e la ricezione di credenziali temporanee. L'utilizzo di IAM Roles Anywhere evita la necessità di ruotare queste credenziali, poiché le credenziali a lungo termine non vengono più memorizzate nell'ambiente on-premises. È necessario monitorare e ruotare il certificato X.509 quando si avvicina alla scadenza.

Risorse

Best practice correlate:

- SEC02-BP02 Utilizzo di credenziali temporanee
- SEC02-BP03 Archiviazione e utilizzo dei segreti in modo sicuro

Documenti correlati:

- Nozioni di base su Gestione dei segreti AWS
- Best practice di IAM
- Identity Providers and Federation
- Soluzioni dei partner per la sicurezza: accesso e controllo degli accessi
- Temporary Security Credentials
- Recupero dei report delle credenziali per l'Account AWS

Video correlati:

- Best Practices for Managing, Retrieving, and Rotating Secrets at Scale
- Managing user permissions at scale with AWS IAM Identity Center
- Mastering identity at every layer of the cake

SEC02-BP06 Impiego dei gruppi di utenti e degli attributi

Definire le autorizzazioni in base a gruppi di utenti e attributi aiuta a ridurre numero e complessità delle policy, semplificando il raggiungimento del principio del privilegio minimo. Puoi usare i gruppi di utenti per gestire le autorizzazioni di molte persone in un'unica posizione, in base alla funzione svolte nell'organizzazione. Gli attributi, come il reparto, il progetto o la posizione, possono fornire un ulteriore livello di portata dei permessi quando le persone svolgono una funzione simile ma per sottoinsiemi diversi di risorse.

Risultato desiderato: puoi applicare modifiche alle autorizzazioni in base alla funzione per tutti gli utenti che la eseguono. L'appartenenza al gruppo e gli attributi regolano le autorizzazioni degli utenti, riducendo la necessità di gestire le autorizzazioni a livello di singolo utente. I gruppi e gli attributi definiti nel gestore dell'identità digitale vengono propagati automaticamente agli ambienti AWS.

Anti-pattern comuni:

- Gestione delle autorizzazioni per singoli utenti e duplicazione tra più utenti.
- Definizione dei gruppi a un livello troppo alto, concessione di autorizzazioni troppo estese.
- Definizione di gruppi a un livello troppo granulare, che crea duplicazioni e confusione sull'appartenenza.
- Utilizzo di gruppi con autorizzazioni duplicate su sottoinsiemi di risorse quando è possibile utilizzare invece gli attributi.
- Nessuna gestione di gruppi, attributi e appartenenze attraverso un gestore dell'identità digitale standardizzato e integrato con gli ambienti AWS.
- Utilizzo della concatenazione dei ruoli quando si utilizzano le sessioni di Centro identità AWS IAM

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Le autorizzazioni AWS sono definite nei documenti denominati policy, associati a un principale, ad esempio un utente, gruppo, ruolo o risorsa. Puoi scalare la gestione delle autorizzazioni organizzando le assegnazioni delle autorizzazioni (gruppo, autorizzazioni, account) in base alla funzione lavorativa, al carico di lavoro e all'ambiente SDLC. Per la forza lavoro, ciò consente di definire i gruppi in base alla funzione svolta dagli utenti per l'organizzazione, anziché in base alle risorse a cui si accede. Ad esempio, un gruppo WebAppDeveloper può avere una policy collegata per la configurazione di servizi come Amazon CloudFront all'interno di un account di sviluppo. Un gruppo `AutomationDeveloper` può avere alcune autorizzazioni che si sovrappongono a quelle del gruppo `WebAppDeveloper`. Queste autorizzazioni comuni possono essere acquisite in una policy separata e associate a entrambi i gruppi, anziché avere utenti di entrambe le funzioni che appartengono a un gruppo `CloudFrontAccess`.

Oltre ai gruppi, è possibile utilizzare gli attributi per un ulteriore ambito dell'accesso. Ad esempio, è possibile avere un attributo Project per gli utenti del gruppo `WebAppDeveloper`, per limitare l'accesso alle risorse specifiche del loro progetto. L'uso di questa tecnica elimina la necessità di avere gruppi diversi per gli sviluppatori di applicazioni che lavorano su progetti diversi, se le loro autorizzazioni sono comunque le stesse. Il modo in cui si fa riferimento agli attributi nelle policy di autorizzazione si basa sulla loro origine, indipendentemente dal fatto che siano definiti come parte del protocollo di federazione (come SAML, OIDC o SCIM), come asserzioni SAML personalizzate o impostati all'interno del Centro identità IAM.

Passaggi dell'implementazione

1. Stabilisci dove definire gruppi e attributi:
   a. Seguendo le indicazioni riportate inSEC02-BP04 Fai affidamento su un gestore dell'identità digitale centralizzato, puoi determinare se occorre definire gruppi e attributi all'interno del gestore dell'identità digitale, all'interno del Centro identità IAM o utilizzare i gruppi di utenti IAM in un account specifico.
2. Definisci i gruppi:
   a. Determina i tuoi gruppi in base alla funzione e all'ambito di accesso richiesti. Valuta se utilizzare una struttura gerarchica o di convenzioni di denominazione per organizzare i gruppi in modo efficace.
   b. Se procedi alla definizione all'interno del Centro identità IAM, crea i gruppi e associa il livello di accesso desiderato utilizzando i set di autorizzazioni.
   c. Se definisci all'interno di un gestore dell'identità digitale esterno, determina se il gestore supporta il protocollo SCIM e valuta la possibilità di abilitare il provisioning automatico all'interno del Centro identità IAM. Questa funzionalità sincronizza la creazione, l'appartenenza e l'eliminazione di gruppi tra il tuo gestore e il Centro identità IAM.
3. Definisci gli attributi:
   a. Se utilizzi un gestore dell'identità digitale esterno, entrambi i protocolli SCIM e SAML 2.0 forniscono determinati attributi per impostazione predefinita. È possibile definire attributi aggiuntivi e trasferirli mediante le asserzioni SAML con il nome dell'attributo `https://aws.amazon.com/SAML/Attributes/PrincipalTag`. Consulta la documentazione del gestore dell'identità digitale per le istruzioni sulla definizione e la configurazione di attributi personalizzati.
   b. Se definisci i ruoli all'interno di Centro identità IAM, abilita la funzionalità di controllo degli accessi basato su attributi (ABAC) e definisci gli attributi come desiderato. Considera gli attributi che si allineano alla struttura dell'organizzazione o alla strategia di tagging delle risorse.

Se richiedi il concatenamento dei ruoli IAM da ruoli IAM assunti tramite Centro identità IAM, i valori come `source-identity` e `principal-tags` non si propagano. Per ulteriori dettagli, consulta Enable and configure attributes for access control.

1. Autorizzazioni di ambito basate su gruppi e attributi:
   a. Prendi in considerazione la possibilità di includere nelle tue policy di autorizzazione condizioni che confrontino gli attributi del tuo principale con gli attributi delle risorse a cui si accede. Ad

esempio, puoi definire una condizione che consenta l'accesso a una risorsa solo se il valore di una chiave di condizione `PrincipalTag` corrisponde a quello di una chiave `ResourceTag` con lo stesso nome.

b. Per la definizione delle policy ABAC, segui le indicazioni contenute nelle best practice e negli esempi relativi alle autorizzazioni ABAC.

c. Rivedi e aggiorna regolarmente la struttura dei gruppi e degli attributi in base all'evoluzione delle esigenze dell'organizzazione per garantire una gestione ottimale delle autorizzazioni.

Risorse

Best practice correlate:

- SEC02-BP04 Fai affidamento su un gestore dell'identità digitale centralizzato
- SEC03-BP02 Concessione dell'accesso con privilegio minimo
- COST02-BP04 Implementazione di gruppi e ruoli

Documenti correlati:

- Best practice di IAM
- Manage Identities in IAM Identity Center
- What Is ABAC for AWS?
- ABAC In IAM Identity Center
- ABAC Policy Examples

Video correlati:

- Managing user permissions at scale with AWS IAM Identity Center
- Mastering identity at every layer of the cake

SEC 3. Come si gestiscono le autorizzazioni per persone e macchine?

Gestisci le autorizzazioni per controllare l'accesso alle identità di persone e macchine che richiedono l'accesso ad AWS e ai tuoi carichi di lavoro. Le autorizzazioni consentono di controllare chi può accedere a cosa e a quali condizioni. Impostando le autorizzazioni per specifiche identità umane e di

macchine, si concede loro l'accesso a determinate azioni di servizio su risorse specifiche. Inoltre, è possibile specificare le condizioni che devono essere vere per concedere l'accesso.

Best practice

- SEC03-BP01 Definizione dei requisiti di accesso
- SEC03-BP02 Concessione dell'accesso con privilegio minimo
- SEC03-BP03 Determinazione di un processo per l'accesso di emergenza
- SEC03-BP04 Riduzione delle autorizzazioni in modo continuo
- SEC03-BP05 Definizione dei guardrail per le autorizzazioni dell'organizzazione
- SEC03-BP06 Gestione degli accessi in base al ciclo di vita
- SEC03-BP07 Analisi dell’accesso multi-account e pubblico
- SEC03-BP08 Condivisione delle risorse in modo sicuro all'interno dell'organizzazione
- SEC03-BP09 Condivisione sicura delle risorse con terze parti

SEC03-BP01 Definizione dei requisiti di accesso

Ogni componente o risorsa del carico di lavoro deve essere accessibile da amministratori, utenti finali o altri componenti. Definisci chiaramente chi o cosa deve avere accesso a ciascun componente e scegli il tipo di identità e il metodo di autenticazione e autorizzazione appropriati.

Anti-pattern comuni:

- Codifica fissa o archiviazione dei segreti nell’applicazione.
- Concessione di autorizzazioni personalizzate per ogni utente.
- Utilizzo di credenziali di lunga durata.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Ogni componente o risorsa del carico di lavoro deve essere accessibile da amministratori, utenti finali o altri componenti. Definisci chiaramente chi o cosa deve avere accesso a ciascun componente e scegli il tipo di identità e il metodo di autenticazione e autorizzazione appropriati.

L’accesso regolare agli Account AWS all’interno di un’organizzazione dovrebbe essere fornito utilizzando l’accesso federato o un gestore dell’identità digitale centralizzato. Occorre anche centralizzare la gestione delle identità e garantire la presenza di una procedura consolidata

per integrare l'accesso ad AWS nel ciclo di vita dell'accesso dei dipendenti. Ad esempio, se un dipendente passa a un ruolo lavorativo con un livello di accesso diverso, anche la sua appartenenza al gruppo deve cambiare per riflettere i nuovi requisiti di accesso.

Nel definire i requisiti di accesso per le identità non umane, determina quali applicazioni e componenti devono accedere, nonché le modalità di concessione delle autorizzazioni. L'utilizzo di ruoli IAM creati con il modello di accesso con privilegio minimo è un approccio consigliato. AWS Le policy gestite forniscono le policy IAM predefinite che coprono la maggior parte dei casi d'uso comuni.

I servizi AWS, come Gestione dei segreti AWS e l'archivio dei parametri AWS Systems Manager consentono di separare i segreti dall'applicazione o dal carico di lavoro in modo sicuro. In Secrets Manager, puoi adottare la rotazione automatica delle credenziali. Puoi usare Systems Manager per fare riferimento a parametri negli script, comandi, documenti SSM, configurazione e flussi di lavoro di automazione utilizzando il nome univoco specificato al momento della creazione del parametro.

Puoi utilizzare AWS IAM Roles Anywhere per ottenere credenziali di sicurezza temporanee in IAM per carichi di lavoro eseguiti all'esterno di AWS. I tuoi carichi di lavoro possono utilizzare le stesse policy IAM e gli stessi ruoli IAM che utilizzi con le applicazioni AWS per accedere alle risorse AWS.

Ove possibile, prediligi le credenziali temporanee a breve termine rispetto a quelle statiche a lungo termine. Per gli scenari in cui occorrono utenti con accesso programmatico e credenziali a lungo termine, usa le ultime informazioni usate per la chiave di accesso per la rotazione e la rimozione delle chiavi di accesso.

Gli utenti hanno bisogno di un accesso programmatico se desiderano interagire con AWS esternamente a Console di gestione AWS. La modalità con cui concedere l'accesso programmatico dipende dal tipo di utente che accede ad AWS.

Per fornire agli utenti l'accesso programmatico, scegli una delle seguenti opzioni.

Quale utente necessita dell'accesso programmatico?	Per	Come
IAM	(Consigliato) Utilizza credenziali della console come credenziali temporanee per firmare richieste programma	Segui le istruzioni per l'interfaccia che desideri utilizzare. • Per AWS CLI, consulta Accesso allo sviluppo locale

Quale utente necessita dell'accesso programmatico?	Per	Come
	tiche alla AWS CLI, agli SDK AWS o alle API AWS.	di AWS nella Guida per l'utente di AWS Command Line Interface. • Per gli SDK AWS, consulta Accesso per lo sviluppo AWS locale nella Guida di riferimento agli SDK e agli strumenti AWS.
Identità della forza lavoro (Utenti gestiti nel centro identità IAM)	Utilizza credenziali temporane e per firmare richieste programmatiche alla AWS CLI, agli SDK AWS o alle API AWS.	Segui le istruzioni per l'interfa ccia che desideri utilizzare. • Per la AWS CLI, consulta la pagina Configurazione della AWS CLI per l'uso di AWS IAM Identity Center nella Guida per l'utente dell'AWS Command Line Interface. • Per gli SDK AWS, gli strumenti e le API AWS, consulta la pagina Autentica zione Centro identità IAM nella Guida di riferimento per SDK e strumenti AWS.
IAM	Utilizza credenziali temporane e per firmare richieste programmatiche alla AWS CLI, agli SDK AWS o alle API AWS.	Segui le istruzioni in Utilizzo di credenziali temporanee con le risorse AWS nella Guida per l'utente IAM.

Quale utente necessita dell'accesso programmatico?	Per	Come
IAM	(Non consigliato) Utilizza credenziali a lungo termine per firmare richieste programmatiche alla AWS CLI, agli SDK AWS o alle API AWS.	Segui le istruzioni per l'interfaccia che desideri utilizzare. • Per la AWS CLI, consulta la pagina Autenticazione tramite credenziali utente IAM nella Guida per l'utente dell'AWS Command Line Interface. • Per gli SDK e gli strumenti AWS, consulta la pagina Autenticazione con credenziali a lungo termine nella Guida di riferimento per SDK e strumenti AWS. • Per le API AWS, consulta la pagina Gestione delle chiavi di accesso per utenti IAM nella Guida per l'utente IAM.

Risorse

Documenti correlati:

- Controllo degli accessi basato su attributi (ABAC)
- AWS IAM Identity Center
- IAM Roles Anywhere
- AWS Managed policies for IAM Identity Center
- AWS Condizioni delle policy IAM
- Casi d'uso IAM
- Rimuovere credenziali non necessarie
- Lavorare con le policy

- How to control access to AWS resources based on Account AWS, OU, or organization
- Identify, arrange, and manage secrets easily using enhanced search in Gestione dei segreti AWS

Video correlati:

- Become an IAM Policy Master in 60 Minutes or Less
- Separation of Duties, Least Privilege, Delegation, and CI/CD
- Streamlining identity and access management for innovation

SEC03-BP02 Concessione dell'accesso con privilegio minimo

Concedi solo l'accesso richiesto dagli utenti per eseguire azioni specifiche su determinate risorse in condizioni particolari. Affidati a gruppi e attributi di identità per impostare in modo dinamico le autorizzazioni su vasta scala, anziché definire le autorizzazioni per i singoli utenti. Ad esempio, puoi concedere a un gruppo di sviluppatori le autorizzazioni per gestire solo le risorse del loro progetto. In questo modo, se uno sviluppatore lascia il progetto, l'accesso viene revocato in automatico senza modificare le policy di accesso sottostanti.

Risultato desiderato: gli utenti dispongono solo delle autorizzazioni minime richieste per le funzioni lavorative specifiche. Utilizzi Account AWS separati per isolare gli sviluppatori dagli ambienti di produzione. Quando gli sviluppatori devono accedere agli ambienti di produzione per attività specifiche, viene concesso un accesso limitato e controllato solo per la durata di tali attività. L'accesso alla produzione viene immediatamente revocato al termine del lavoro necessario. Esegui revisioni regolari delle autorizzazioni e revocale prontamente quando non sono più necessarie, ad esempio quando un utente cambia ruolo o lascia l'organizzazione. Limita i privilegi di amministratore a un gruppo ristretto e attendibile per ridurre l'esposizione al rischio. Assegna agli account di computer o di sistema solo le autorizzazioni minime necessarie per eseguire le attività previste.

Anti-pattern comuni:

- Per impostazione predefinita, concedi agli utenti le autorizzazioni di amministratore.
- Utilizzi l'account utente root per le attività quotidiane.
- Crei policy eccessivamente permissive senza un ambito adeguato.
- Le revisioni delle autorizzazioni sono rare, il che porta all'insinuarsi di autorizzazioni.
- Per l'isolamento dell'ambiente o la gestione delle autorizzazioni, fai affidamento esclusivamente al controllo degli accessi basato su attributi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Secondo il principio del privilegio minimo, le identità dovrebbero essere autorizzate a eseguire solo il più piccolo insieme di azioni necessarie per lo svolgimento di un'attività specifica. In questo modo usabilità, efficienza e sicurezza sono bilanciate. Seguendo questo principio si limitano gli accessi indesiderati e si può monitorare chi accede a quali risorse. Per impostazione predefinita, ruoli e utenti IAM non dispongono di autorizzazioni. Per impostazione predefinita, l'utente root dispone dell'accesso completo e deve essere strettamente controllato, monitorato e utilizzato solo per le attività che richiedono l'accesso root.

Le policy IAM consentono di concedere in modo esplicito le autorizzazioni ai ruoli IAM o a risorse specifiche. Ad esempio, le policy basate su identità possono essere collegate ai gruppi IAM, mentre i bucket S3 possono essere controllati da policy basate su risorse.

Quando crei una policy IAM, puoi specificare le azioni di servizio, le risorse e le condizioni che devono essere vere affinché AWS consenta o rifiuti l'accesso. AWS supporta una serie di condizioni per aiutare a ridurre l'ambito dell'accesso. Ad esempio, con la chiave di condizione PrincipalOrgID, puoi negare azioni se il richiedente non fa parte della tua organizzazione AWS.

Puoi anche controllare le richieste effettuate dai servizi AWS per tuo conto, ad esempio AWS CloudFormation per la creazione di una funzione AWS Lambda, utilizzando la chiave di condizione CalledVia. Puoi stratificare diversi tipi di policy per stabilire una difesa in profondità e limitare le autorizzazioni complessive degli utenti. Puoi anche limitare le autorizzazioni che possono essere concesse e le relative condizioni. Ad esempio, puoi consentire ai team del carico di lavoro di creare le proprie policy IAM per i sistemi che realizzano, ma solo se applicano un limite delle autorizzazioni per circoscrivere le autorizzazioni massime che possono essere concesse.

Passaggi dell'implementazione

- Implementa policy con privilegio minimo: assegna policy di accesso con privilegio minimo a ruoli e gruppi IAM in modo da rispecchiare il ruolo o la funzione dell'utente che hai definito.
- Isola gli ambienti di sviluppo e produzione tramite Account AWS separati: utilizza Account AWS separati per gli ambienti di sviluppo e di produzione e controlla l'accesso tra di essi utilizzando policy di controllo dei servizi, policy delle risorse e policy identità.
- Policy di base sull'utilizzo delle API: un modo per determinare le autorizzazioni necessarie consiste nel rivedere i log AWS CloudTrail. Puoi utilizzare questa revisione per creare autorizzazioni personalizzate in base alle azioni che l'utente esegue effettivamente all'interno di AWS. IAM

Access Analyzer può generare automaticamente una policy IAM basata su attività di accesso. Puoi usare IAM Access Advisor a livello di organizzazione o account per tenere traccia delle ultime informazioni a cui si ha avuto accesso per una particolare policy.

- Prendi in considerazione l'utilizzo di policy gestite da AWS per funzioni lavorative: quando inizi a creare policy di autorizzazioni granulari, può essere utile utilizzare policy gestite da AWS per ruoli lavorativi comuni, ad esempio contabili, amministratori di database e data scientist. Questi policy possono aiutare a restringere l'accesso degli utenti mentre si determina come implementare i criteri di privilegio minimo.
- Rimuovi le autorizzazioni non necessarie: rileva e rimuovi le entità, le credenziali e le autorizzazioni IAM non utilizzate per ottenere il principio del privilegio minimo. Puoi utilizzare Sistema di analisi degli accessi AWS IAM per identificare gli accessi esterni e quelli non utilizzati e la generazione di policy del Sistema di analisi degli accessi AWS IAM può aiutare a eseguire il fine-tuning delle policy di autorizzazione.
- Assicurati che gli utenti abbiano un accesso limitato agli ambienti di produzione: gli utenti devono avere accesso agli ambienti di produzione solo in presenza di un caso d'uso valido. Una volta eseguite le attività specifiche che richiedono l'accesso alla produzione, l'accesso dell'utente deve essere revocato. Limitare l'accesso agli ambienti di produzione contribuisce a evitare eventi indesiderati con impatto sulla produzione e contiene gli effetti di accessi involontari.
- Considera i confini delle autorizzazioni: un limite delle autorizzazioni è una funzionalità per l'utilizzo di una policy gestita che stabilisce le autorizzazioni massime che una policy basata sull'identità può concedere a un'entità IAM. Il limite delle autorizzazioni di un'entità consente di eseguire solo le operazioni consentite dalle sue policy basate su identità e dai suoi limiti delle autorizzazioni.
- Perfeziona l'accesso usando il controllo dell'accesso basato sugli attributi e i tag delle risorse Il controllo degli accessi basato su attributi (ABAC) usando i tag delle risorse può essere usato per perfezionare le autorizzazioni quando è supportato. Puoi utilizzare un modello ABAC che confronta i tag dei principali con i tag delle risorse per perfezionare l'accesso in base a dimensioni personalizzate definite dall'utente. Questo approccio può semplificare e ridurre il numero di policy di autorizzazione nell'organizzazione.
  - Si consiglia di utilizzare ABAC per il controllo degli accessi solo quando sia i principali che le risorse sono di proprietà dell'organizzazione AWS. Le parti esterne possono utilizzare gli stessi nomi e valori di tag dell'organizzazione per i propri principali e risorse. Se fai affidamento esclusivamente su queste coppie nome-valore per concedere l'accesso a risorse o principali esterni, potresti fornire autorizzazioni indesiderate.
- Utilizza le policy di controllo dei servizi per AWS Organizations: le policy di controllo dei servizi controllano centralmente le autorizzazioni massime disponibili per gli account dei membri

dell'organizzazione. È importante notare che puoi utilizzare le policy di controllo dei servizi per limitare le autorizzazioni dell'utente root negli account membri. Prendi anche in considerazione la possibilità di usare AWS Control Tower, che offre controlli gestiti prescrittivi che arricchiscono AWS Organizations. Puoi anche definire i tuoi controlli in Control Tower.

- Stabilisci una policy del ciclo di vita degli utenti per la tua organizzazione: le policy del ciclo di vita degli utenti definiscono le attività da eseguire in caso di onboarding degli utenti in AWS, cambiamento di ruolo o ambito lavorativo o cessata necessità di accedere a AWS. Esegui revisioni delle autorizzazioni durante ogni fase del ciclo di vita di un utente per verificare che le autorizzazioni siano adeguatamente restrittive e per evitare l'insorgere di autorizzazioni.
- Stabilisci una pianificazione regolare per rivedere le autorizzazioni e rimuovere quelle non necessarie: è necessario rivedere regolarmente l'accesso utente per verificare che non sia eccessivamente permissivo. AWS Config e Sistema di analisi degli accessi AWS IAM possono essere d'aiuto durante gli audit delle autorizzazioni degli utenti.
- Stabilisci una matrice dei ruoli professionali: una matrice dei ruoli professionali visualizza i vari ruoli e i livelli di accesso richiesti all'interno della tua impronta AWS. Con una matrice dei ruoli professionali puoi definire e separare le autorizzazioni in base alle responsabilità degli utenti all'interno dell'organizzazione. Utilizza i gruppi anziché applicare le autorizzazioni direttamente a singoli utenti o ruoli.

Risorse

Documenti correlati:

- Grant least privilege
- Permissions boundaries for IAM entities
- Techniques for writing least privilege IAM policies
- IAM Access Analyzer makes it easier to implement least privilege permissions by generating IAM policies based on access activity
- Delegate permission management to developers by using IAM permissions boundaries
- Perfezionare le autorizzazioni utilizzando le informazioni dell'ultimo accesso
- IAM policy and when to use them
- Test delle policy IAM con il simulatore di policy IAM
- Guardrail in AWS Control Tower
- Zero Trust architectures: An AWS perspective

- How to implement the principle of least privilege with CloudFormation StackSets
- Controllo degli accessi basato su attributi (ABAC)
- Reducing policy scope by viewing user activity
- View role access
- Uso dei tag per organizzare il proprio ambiente e aumentare la responsabilità
- Strategie di applicazione di tag AWS
- Applicazione di tag alle risorse AWS

Video correlati:

- Next-generation permissions management
- Zero Trust: An AWS perspective

SEC03-BP03 Determinazione di un processo per l'accesso di emergenza

Crea un processo che consenta l'accesso di emergenza ai tuoi carichi di lavoro nell'improbabile eventualità che si verifichi un problema con il tuo gestore dell'identità digitale centralizzato.

Devi progettare processi per diverse modalità di guasto che potrebbero causare un evento di emergenza. Ad esempio, in circostanze normali, gli utenti della tua forza lavoro si federano nel cloud utilizzando un gestore dell'identità digitale centralizzato (SEC02-BP04) per gestire i loro carichi di lavoro. Tuttavia, se il tuo gestore dell'identità digitale centralizzato riscontra un errore o la configurazione per la federazione nel cloud subisce modifiche, gli utenti della tua forza lavoro potrebbero non essere in grado di federarsi nel cloud. Un processo di accesso di emergenza consente agli amministratori autorizzati di accedere alle risorse cloud tramite mezzi alternativi (come una forma alternativa di federazione o l'accesso diretto degli utenti) per risolvere problemi relativi alla configurazione della federazione o ai carichi di lavoro. Si ricorre al processo di accesso di emergenza fino al ripristino del normale meccanismo di federazione.

Risultato desiderato:

- Hai definito e documentato le modalità di guasto che costituiscono un'emergenza: considera le circostanze normali e i sistemi da cui dipendono gli utenti per gestire i loro carichi di lavoro. Prendi in considerazione quali guasti possono interessare ciascuna di queste dipendenze e causare una situazione di emergenza. Potresti trovare utili le domande e le best practice del pilastro

dell'affidabilità per individuare le modalità di errore e progettare sistemi più resilienti al fine di ridurre al minimo la probabilità di guasti.

- Hai documentato i passaggi da seguire per confermare che un guasto costituisce un'emergenza. Ad esempio, puoi richiedere agli amministratori di identità di controllare lo stato dei gestori delle identità digitali primari e di standby e, se entrambi non sono disponibili, dichiarare un evento di emergenza per guasto del gestore dell'identità digitale.
- È stato definito un processo di accesso di emergenza specifico per ogni tipo di modalità di emergenza o di guasto. Essere specifici può ridurre la tentazione da parte degli utenti di abusare di un processo generale per tutti i tipi di emergenze. I processi di accesso di emergenza illustrano le circostanze in cui ciascun processo va o non va utilizzato e indicano processi alternativi applicabili.
- I tuoi processi sono ben documentati con istruzioni e playbook dettagliati, facili da mettere in pratica in modo rapido ed efficiente. Ricorda che un evento di emergenza può essere un momento stressante per i tuoi utenti, che potrebbero essere sotto pressione per motivi di tempo, quindi progetta il tuo processo in modo che sia il più semplice possibile.

Anti-pattern comuni:

- Non si dispone di procedure di accesso di emergenza ben documentate e collaudate. Gli utenti non sono preparati per un'emergenza e seguono processi improvvisati quando si verifica un evento di emergenza.
- I processi di accesso di emergenza dipendono dagli stessi sistemi (come un gestore dell'identità digitale centralizzato) dei normali meccanismi di accesso. Ciò significa che il guasto di un sistema di questo tipo può influire sui normali meccanismi di accesso e di emergenza e compromettere la capacità di ripristino dall'errore.
- I processi di accesso di emergenza vengono utilizzati in situazioni non di emergenza. Ad esempio, gli utenti utilizzano spesso in modo improprio i processi di accesso di emergenza poiché trovano più facile apportare modifiche direttamente piuttosto che inviarle tramite una pipeline.
- I processi di accesso di emergenza non generano log sufficienti per effettuare l'audit dei processi oppure i log non vengono monitorati per segnalare un potenziale uso improprio dei processi.

Vantaggi dell'adozione di questa best practice:

- Grazie a processi di accesso di emergenza ben documentati e collaudati, puoi ridurre il tempo impiegato dagli utenti per rispondere a un evento di emergenza e risolverlo. Ciò può comportare una riduzione dei tempi di inattività e una maggiore disponibilità dei servizi forniti ai clienti.

- È possibile tenere traccia di ogni richiesta di accesso di emergenza e rilevare e segnalare i casi di tentativi non autorizzati di uso improprio del processo per eventi non di emergenza.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

La presente sezione fornisce indicazioni per la creazione di processi di accesso di emergenza per diverse modalità di errore relative ai carichi di lavoro implementati su AWS, a partire da linee guida comuni applicabili a tutte le modalità di errore fino a linee guida specifiche in base al tipo di errore.

Linee guida comuni per tutte le modalità di errore

Nella progettazione di un processo di accesso di emergenza per una modalità di errore, tieni presente quanto segue:

- Documenta prerequisiti e presupposti del processo: quando il processo deve e non deve essere utilizzato. Aiuta a descrivere in dettaglio la modalità di errore e a documentare le ipotesi, come lo stato di altri sistemi correlati. Ad esempio, il processo per la modalità di errore 2 presuppone che il gestore dell'identità digitale sia disponibile, ma la configurazione in AWS è stata modificata o è scaduta.
- Crea preliminarmente le risorse necessarie per il processo di accesso di emergenza (SEC10-BP05). Ad esempio, crea preliminarmente l'accesso di emergenza a un Account AWS con ruoli e utenti IAM e in tutti gli account del carico di lavoro creando ruoli IAM multi-account. Ciò assicura che queste risorse siano pronte e disponibili quando si verifica un evento di emergenza. Creando in modo preliminare le risorse, non si dipende dalle API del piano di controllo (control-plane) AWS (utilizzate per creare e modificare risorse AWS) che potrebbero non essere disponibili in caso di emergenza. Inoltre, creando in modo preliminare le risorse IAM, non è necessario tenere conto dei potenziali ritardi dovuti all'eventuale consistenza.
- Includi i processi di accesso di emergenza nei tuoi piani di gestione degli incidenti (SEC10-BP02). Documenta le modalità in cui si tiene traccia degli eventi di emergenza e come questi vengono comunicati ad altri membri dell'organizzazione, come i team di pari livello, la leadership e, se applicabile, esternamente ai clienti e ai partner aziendali.
- Definisci il processo di richiesta di accesso di emergenza nel tuo sistema di flusso di lavoro esistente, se ne hai uno, per le richieste di assistenza. In genere, tali sistemi di flusso di lavoro consentono di creare moduli di acquisizione per raccogliere informazioni sulla richiesta, tenere traccia della richiesta in ogni fase del flusso di lavoro e aggiungere passaggi di approvazione

automatici e manuali. Collega ciascuna richiesta a un evento di emergenza corrispondente tracciato nel tuo sistema di gestione degli incidenti. Disporre di un sistema uniforme per gli accessi di emergenza consente di tenere traccia di tali richieste in un unico sistema, analizzare le tendenze di utilizzo e migliorare i processi.

- Verifica che i processi di accesso di emergenza possano essere avviati solo da utenti autorizzati e richiedano l'approvazione di colleghi o manager dell'utente, a seconda dei casi. Il processo di approvazione deve funzionare in modo efficacie sia all'interno sia al di fuori dell'orario lavorativo. Definisci in che modo le richieste di approvazione possono essere eseguite da approvatori secondari, qualora gli approvatori principali non fossero disponibili, e come vengono inoltrate lungo la catena di gestione fino all'approvazione.
- Implementa affidabili meccanismi di registrazione dei log, monitoraggio e avviso per il processo e i meccanismi di accesso di emergenza. Genera log di audit dettagliati per tutti i tentativi riusciti e non riusciti di ottenere l'accesso di emergenza. Metti in correlazione l'attività con gli eventi di emergenza in corso dal sistema di gestione degli incidenti e attiva gli avvisi quando le azioni si verificano al di fuori dei periodi previsti o quando l'account di accesso di emergenza viene utilizzato durante le normali operazioni. L'account di accesso di emergenza deve essere utilizzato solo in caso di emergenza, poiché le procedure break-glass possono essere considerate una backdoor. Effettua l'integrazione con lo strumento di gestione delle informazioni e degli eventi di sicurezza (SIEM) o con AWS Security Hub CSPM per segnalare e verificare tutte le attività durante il periodo di accesso di emergenza. Quando torni alla normale operatività, effettua la rotazione automatica delle credenziali di accesso di emergenza e informa i team interessati.
- Testa periodicamente i processi di accesso di emergenza per verificare che i passaggi siano chiari e garantire il livello di accesso corretto in modo rapido ed efficiente. I processi di accesso di emergenza devono essere testati nell'ambito delle simulazioni di risposta agli incidenti (SEC10-BP07) e dei test di disaster recovery (REL13-BP03).

Modalità di errore 1: il gestore dell'identità digitale utilizzato per la federazione dell'accesso ad AWS non è disponibile

Come illustrato in SEC02-BP04 Fai affidamento su un gestore dell'identità digitale centralizzato, ti consigliamo di affidarti a un gestore dell'identità digitale centralizzato per federare gli utenti della tua forza lavoro e garantire loro l'accesso agli Account AWS. È possibile federare l'accesso a più Account AWS all'interno dell'organizzazione AWSutilizzando il Centro identità IAM oppure federare l'accesso individuale agli Account AWS utilizzando IAM. In entrambi i casi, gli utenti della forza lavoro si autenticano con il gestore dell'identità digitale centralizzato prima di essere reindirizzati a un endpoint di accesso AWS per l'autenticazione unica.

Nell'improbabile eventualità che il gestore dell'identità digitale centralizzato non sia disponibile, gli utenti della tua forza lavoro non possono federarsi per accedere agli Account AWS o gestire i propri carichi di lavoro. In questo evento di emergenza, puoi fornire un processo di accesso di emergenza secondo cui un piccolo gruppo di amministratori può accedere agli Account AWS per eseguire attività urgenti per le quali non è possibile attendere che i tuoi gestori delle identità digitali centralizzati tornino online. Ad esempio, il tuo gestore dell'identità digitale non è disponibile per 4 ore e durante quel periodo devi modificare i limiti massimi di un gruppo Amazon EC2 Auto Scaling in un account di produzione per gestire un picco imprevisto nel traffico dei clienti. Gli amministratori di emergenza devono seguire la procedura di accesso di emergenza per accedere a un Account AWS di produzione specifico e apportare le modifiche necessarie.

Il processo di accesso di emergenza si basa su un accesso di emergenza a un Account AWS creato preliminarmente, utilizzato esclusivamente per questo tipo di accessi e dispone di risorse AWS (come ruoli e utenti IAM) per supportare il processo di accesso di emergenza. Durante le normali operazioni, nessuno deve accedere all'account di accesso di emergenza ed è necessario monitorare e fornire avvisi riguardo a usi impropri di questo account (per maggiori dettagli, vedi la sezione precedente Linee guida comuni).

L'account di accesso di emergenza dispone di ruoli IAM di accesso di emergenza con autorizzazioni per assumere ruoli multi-account negli Account AWS che richiedono l'accesso di emergenza. Questi ruoli IAM sono creati preliminarmente e configurati con policy di attendibilità che valutano i ruoli IAM dell'account di emergenza come attendibili.

Per il processo di accesso di emergenza è possibile utilizzare uno dei seguenti approcci:

- Puoi creare in modo preliminare un set di utenti IAM per gli amministratori di emergenza nell'account di accesso di emergenza con password complesse e token MFA associati. Tali utenti IAM dispongono delle autorizzazioni per assumere i ruoli IAM che consentono l'accesso multi-account all'Account AWS per cui è richiesto l'accesso di emergenza. Ti consigliamo di creare il minor numero possibile di utenti di questo tipo e di assegnare ogni utente a un unico amministratore di emergenza. Durante un'emergenza, un utente amministratore di emergenza accede all'account di accesso di emergenza utilizzando la propria password e il codice token MFA, passa al ruolo IAM di accesso di emergenza nell'account di emergenza e infine passa al ruolo IAM di accesso di emergenza nell'account del carico di lavoro per eseguire l'azione di modifica di emergenza. Il vantaggio di questo approccio è che ogni utente IAM è assegnato a un amministratore di emergenza e puoi sapere quale utente ha effettuato l'accesso esaminando gli eventi CloudTrail. Lo svantaggio è che è necessario mantenere più utenti IAM con le relative password di lunga durata e i token MFA associati.

- Puoi usare l'accesso di emergenza dell'utente root Account AWS per accedere all'account di emergenza, assumere il ruolo IAM per l'accesso di emergenza e poi il ruolo multi-account nell'account del carico di lavoro. È consigliabile impostare una password sicura e più token MFA per l'utente root. Consigliamo inoltre di archiviare la password e i token MFA in un archivio di credenziali aziendali sicuro, che applichi policy di autenticazione e autorizzazione avanzate. Proteggi i fattori di reimpostazione della password e del token MFA: imposta l'indirizzo e-mail dell'account su una lista di distribuzione e-mail monitorata dagli amministratori della sicurezza del cloud e il numero di telefono dell'account su un numero di telefono condiviso anch'esso monitorato dagli amministratori della sicurezza. Il vantaggio di questo approccio è l'esistenza di un solo set di credenziali utente root da gestire. Lo svantaggio è che, trattandosi di un utente condiviso, più amministratori hanno la possibilità di accedere come utente root. Controlla gli eventi del log del tuo vault aziendale per identificare quale amministratore ha utilizzato la password dell'utente root.

Modalità di errore 2: la configurazione del gestore dell'identità digitale in AWS è stata modificata o è scaduta

Per consentire agli utenti della tua forza lavoro di effettuare l'accesso federato agli Account AWS, puoi configurare il Centro identità IAM con un gestore dell'identità digitale esterno o un gestore dell'identità digitale IAM (SEC02-BP04). In genere, la configurazione si effettua importando un documento XML di metadati SAML fornito dal gestore dell'identità digitale. Il documento XML di metadati include un certificato X.509 corrispondente a una chiave privata utilizzata dal gestore dell'identità digitale per firmare le sue asserzioni SAML.

Queste configurazioni lato AWS possono essere modificate o eliminate per errore da un amministratore. In un altro scenario, può accadere che il certificato X.509 importato in AWS sia scaduto e che un nuovo XML di metadati con un nuovo certificato non sia ancora stato importato in AWS. In entrambi gli scenari, la federazione degli utenti della forza lavoro per accedere ad AWS può essere interrotta, creando così una situazione di emergenza.

In un caso di emergenza di questo tipo, puoi fornire agli amministratori delle identità l'accesso ad AWS per risolvere i problemi di federazione. Ad esempio, l'amministratore delle identità utilizza la procedura di accesso di emergenza per accedere a un Account AWS, passa a un ruolo nell'account amministratore del Centro identità e riattiva la federazione aggiornando la configurazione del gestore dell'identità digitale esterno e importando l'ultimo documento XML di metadati SAML rilasciato dal gestore dell'identità digitale. Una volta ristabilita la federazione, gli utenti della forza lavoro continuano a utilizzare il normale processo operativo per federare l'accesso ai propri account di carico di lavoro.

È possibile seguire gli approcci illustrati nella sezione precedente Modalità di errore 1 per creare un processo di accesso di emergenza. Puoi concedere le autorizzazioni con il privilegio minimo agli amministratori delle identità per accedere solo all'account amministratore di Centro identità ed eseguire azioni in Centro identità in quell'account.

Modalità di errore 3: blocco del Centro identità

Nell'improbabile eventualità di un blocco del Centro identità IAM o una Regione AWS, ti consigliamo di eseguire una configurazione per fornire l'accesso temporaneo alla Console di gestione AWS.

Il processo di accesso di emergenza utilizza la federazione diretta rilasciata dal gestore dell'identità digitale a un IAM per accedere a un account di emergenza. Per informazioni dettagliate sul processo e sulle considerazioni di progettazione, consulta Set up emergency access to the Console di gestione AWS.

Passaggi dell'implementazione

Passaggi comuni per tutte le modalità di errore

- Crea un Account AWS dedicato per gli accessi di emergenza. Crea preliminarmente le risorse IAM necessarie nell'account, come i ruoli IAM o gli utenti IAM, e, in modo facoltativo, i gestori delle identità digitali IAM. Inoltre, crea preliminarmente ruoli IAM multi-account negli Account AWS del carico di lavoro dotati di relazioni di fiducia con i ruoli IAM corrispondenti nell'account di accesso di emergenza. Puoi usare CloudFormation StackSets con AWS Organizations per creare tali risorse negli account dei membri della tua organizzazione.

- Crea una policy di controllo dei servizi AWS Organizations per negare l'eliminazione e la modifica dei ruoli IAM multi-account negli Account AWS dei membri.
- Abilita CloudTrail per l'accesso di emergenza a un Account AWS e invia gli eventi di trail a un bucket S3 centrale nella raccolta di log relativa all'Account AWS. Se utilizzi AWS Control Tower per configurare e gestire il tuo ambiente AWS multi-account, ogni account che crei utilizzando AWS Control Tower o a cui ti iscrivi in AWS Control Tower presenta CloudTrail abilitato per impostazione predefinita e viene inviato a un bucket S3 in un Account AWS con archivio di log dedicato.
- Monitora l'attività dell'account di accesso di emergenza creando regole EventBridge coerenti con l'accesso alla console e all'attività dell'API da parte dei ruoli IAM di emergenza. Invia notifiche al tuo centro operativo di sicurezza quando si verificano attività al di fuori di un evento di emergenza in corso e di cui hai traccia nel tuo sistema di gestione degli incidenti.

Passaggi aggiuntivi per la Modalità di errore 1: il gestore dell'identità digitale utilizzato per la federazione dell'accesso ad AWS non è disponibile; per la Modalità di errore 2: la configurazione del gestore dell'identità digitale su AWS è stata modificata o è scaduta

- Crea preliminarmente le risorse in base al meccanismo scelto per l'accesso di emergenza:
  - Usando gli utenti IAM: crea preliminarmente gli utenti IAM con password complesse e dispositivi MFA associati.
  - Utilizzando l'utente utente root dell'account di emergenza: configura l'utente root con una password sicura e archivia la password nel tuo vault di credenziali aziendali. Associa più dispositivi MFA fisici all'utente root e archivia i dispositivi in posizioni a cui i membri del team di amministrazione delle emergenze possono accedere rapidamente.

Passaggi aggiuntivi per la Modalità di errore 3: blocco del Centro identità

- Come illustrato in Set up emergency access to the Console di gestione AWS, per l'accesso di emergenza a un Account AWS, crea un gestore dell'identità digitale IAM per abilitare la federazione SAML diretta dal tuo gestore dell'identità digitale.
- Crea gruppi operativi di emergenza nel tuo IdP senza membri.
- Crea ruoli IAM corrispondenti ai gruppi operativi di emergenza nell'account di accesso di emergenza.

Risorse

Best practice Well-Architected correlate:

- SEC02-BP04 Fai affidamento su un gestore dell'identità digitale centralizzato
- SEC03-BP02 Concessione dell'accesso con privilegio minimo
- SEC10-BP02 Sviluppo di piani di gestione degli incidenti
- SEC10-BP07 Esecuzione di giornate di gioco

Documenti correlati:

- Set up emergency access to the Console di gestione AWS
- Enabling SAML 2,0 federated users to access the Console di gestione AWS
- Break glass access

Video correlati:

- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center
- AWS re:Inforce 2022 - AWS Identity and Access Management (IAM) deep dive

Esempi correlati:

- AWS Break Glass Role
- AWS Framework per playbook per i clienti
- AWS incident response playbook samples

SEC03-BP04 Riduzione delle autorizzazioni in modo continuo

Man mano che i team determinano gli accessi necessari, rimuovi le autorizzazioni non necessarie e stabilisci processi di revisione per ottenere le autorizzazioni con il privilegio minimo. Monitora costantemente e rimuovi le identità e le autorizzazioni inutilizzate per l'accesso sia umano che delle macchine.

Risultato desiderato: le policy di autorizzazione rispettano il principio del privilegio minimo. Man mano che le mansioni e i ruoli vengono definiti meglio, è necessario rivedere le policy di autorizzazione per eliminare le autorizzazioni non necessarie. Questo approccio riduce la portata dell'impatto nel caso di esposizione accidentale delle credenziali o di accesso in altro modo senza autorizzazione.

Anti-pattern comuni:

- L'impostazione predefinita è la concessione delle autorizzazioni di amministratore agli utenti.
- Creazione di policy eccessivamente permissive, ma senza privilegi completi di amministratore.
- Mantenimento delle policy di autorizzazione anche quando non sono più necessarie.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Quando i team e i progetti sono in fase iniziale, è possibile usare policy di autorizzazione permissiva per stimolare l'innovazione e l'agilità. Ad esempio, in un ambiente di sviluppo o di test, gli sviluppatori possono avere accesso a un'ampia gamma di servizi AWS. Si consiglia di valutare in modo costante gli accessi e di limitare l'accesso solo ai servizi e alle azioni di servizio necessari per completare il lavoro in corso. Raccomandiamo questa valutazione sia per l'identità umana che per quella

macchina. Le identità macchina, talvolta chiamate account di sistema o di servizio, sono identità che consentono ad AWS di accedere ad applicazioni o server. Questo accesso è particolarmente importante in un ambiente di produzione, dove autorizzazioni troppo permissive possono avere un ampio impatto e potenzialmente esporre i dati dei clienti.

AWS offre diversi metodi per identificare utenti, ruoli, autorizzazioni e credenziali non utilizzati. AWS può anche aiutare ad analizzare l'attività di accesso di ruoli e utenti IAM, comprese le chiavi di accesso associate, e l'accesso alle risorse AWS, come gli oggetti nei bucket Amazon S3. La generazione di policy di AWS Identity and Access Management Access Analyzer può aiutare a creare policy di autorizzazione restrittive in base ai servizi e alle azioni effettive con cui interagisce un principale. Il controllo degli accessi basato su attributi (ABAC) consente di semplificare la gestione delle autorizzazioni, offrendo la possibilità di fornire le autorizzazioni agli utenti sulla base dei loro attributi anziché allegare le policy di autorizzazione direttamente a ciascun utente.

Passaggi dell’implementazione

- Usa AWS Identity and Access Management Access Analyzer: IAM Access Analyzer aiuta a identificare le risorse dell'organizzazione e gli account, ad esempio i bucket Amazon Simple Storage Service (Amazon S3) o i ruoli IAM, condivisi con un'entità esterna.
- Utilizza la generazione di policy di IAM Access Analyzer: la generazione di policy di IAM Access Analyzer consente di creare policy di autorizzazione granulari basate sull'attività di accesso di ruoli o utenti IAM.
- Esegui il test delle autorizzazioni in ambienti di livello inferiore prima della produzione: inizia utilizzando gli ambienti sandbox e di sviluppo meno critici per testare le autorizzazioni richieste per le varie funzioni lavorative utilizzando Sistema di analisi degli accessi AWS IAM. Quindi, limita e convalida progressivamente queste autorizzazioni negli ambienti di test, controllo qualità e gestione temporanea prima di applicarle in produzione. Gli ambienti di livello inferiore possono avere inizialmente autorizzazioni più permissive, poiché le policy di controllo dei servizi (SCP) applicano dei guardrail limitando il numero massimo di autorizzazioni concesse.
- Determina un periodo di tempo e una policy di utilizzo accettabili per ruoli e utenti IAM: utilizza il timestamp dell'ultimo accesso per identificare utenti e ruoli non utilizzati e rimuoverli. Rivedi le informazioni sull'ultimo accesso al servizio e sull'ultima azione per identificare e definire le autorizzazioni per specifici utenti e ruoli. Ad esempio, puoi utilizzare le informazioni sull'ultimo accesso per identificare le azioni specifiche di Amazon S3 richieste dal ruolo dell'applicazione e delimitare l'accesso del ruolo solo a tali azioni. Le funzionalità relative alle informazioni sull'ultimo accesso sono disponibili nella Console di gestione AWS e consentono di incorporarle in modo programmatico nei flussi di lavoro dell'infrastruttura e negli strumenti automatizzati.

- Prendi in considerazione la possibilità di creare log degli eventi relativi ai dati in AWS CloudTrail: per impostazione predefinita, CloudTrail non crea log degli eventi relativi ai dati come le attività a livello di oggetto di Amazon S3 (ad esempio, `GetObject` e `DeleteObject`) o le attività delle tabelle Amazon DynamoDB (ad esempio `PutItem` e `DeleteItem`). Considera l'uso della creazione di log di questi eventi per stabilire quali utenti e ruoli devono accedere a specifici oggetti Amazon S3 o elementi di tabelle DynamoDB.

Risorse

Documenti correlati:

- Grant least privilege
- Rimuovere credenziali non necessarie
- What is AWS CloudTrail?
- Lavorare con le policy
- Logging and monitoring DynamoDB
- Abilitare la creazione di log di eventi CloudTrail per bucket e oggetti Amazon S
- Recupero dei report delle credenziali per l Account AWS

Video correlati:

- Become an IAM Policy Master in 60 Minutes or Less
- Separation of Duties, Least Privilege, Delegation, and CI/CD
- AWS re:Inforce 2022 - AWS Identity and Access Management (IAM) deep dive

SEC03-BP05 Definizione dei guardrail per le autorizzazioni dell'organizzazione

Utilizza i guardrail delle autorizzazioni per ridurre l'ambito delle autorizzazioni disponibili concedibili ai principali. La catena di valutazione delle policy di autorizzazione comprende i guardrail così da determinare le autorizzazioni effettive di un principale quando adotta decisioni relative alle autorizzazioni. È possibile definire i guardrail utilizzando un approccio basato sui livelli. Applica alcuni guardrail in modo esteso all'intera organizzazione e applicane altri in modo granulare alle sessioni di accesso temporaneo.

Risultato desiderato: hai un chiaro isolamento degli ambienti utilizzando Account AWS separati. Le policy di controllo dei servizi (SCP) consentono di definire i guardrail delle autorizzazioni a

livello di organizzazione. I guardrail più estesi sono impostati ai livelli gerarchici più vicini alla radice dell'organizzazione, mentre i guardrail più rigidi sono impostati più vicino al livello dei singoli account.

Se supportate, le policy sulle risorse definiscono le condizioni che un principale deve soddisfare per ottenere l'accesso a una risorsa. Le policy per le risorse, inoltre, definiscono l'insieme delle azioni consentite, laddove appropriato. I limiti delle autorizzazioni sono posti sui principali che gestiscono le autorizzazioni del carico di lavoro, delegando la gestione delle autorizzazioni ai singoli proprietari del carico di lavoro.

Anti-pattern comuni:

- Creare membri di Account AWS all'interno di un'organizzazione AWS, senza utilizzare SCP per limitare l'uso e le autorizzazioni disponibili alle relative credenziali root.
- Assegnare le autorizzazioni in base al privilegio minimo, senza però porre guardrail sull'insieme massimo di autorizzazioni concedibili.
- Affidarsi alla base di rifiuto implicito di AWS IAM per limitare le autorizzazioni, confidando nel fatto che le policy non concedano un'autorizzazione esplicita indesiderata.
- Eseguire più ambienti di carico di lavoro nello stesso Account AWS e affidarsi quindi a meccanismi come VPC, tag o policy sulle risorse per applicare i limiti delle autorizzazioni.

Vantaggi derivanti dall'adozione di questa best practice: i guardrail di autorizzazione contribuiscono a creare la certezza che le autorizzazioni indesiderate non possano essere concesse, anche quando una policy di autorizzazione tenta di farlo. Ciò può semplificare la definizione e la gestione delle autorizzazioni riducendo l'ambito massimo delle autorizzazioni da prendere in considerazione.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Ti consigliamo di utilizzare un approccio basato sui livelli per definire i guardrail di autorizzazione per la tua organizzazione. Questo approccio riduce in modo sistematico il set massimo di autorizzazioni possibili con l'applicazione di livelli aggiuntivi. Ciò consente di concedere l'accesso in base al principio del privilegio minimo, riducendo il rischio di accessi non intenzionali dovuti a un'errata configurazione delle policy.

Il primo passo per definire i guardrail delle autorizzazioni è isolare i carichi di lavoro e gli ambienti in Account AWS separati. I principali di un account non possono accedere alle risorse di un altro account senza l'autorizzazione esplicita in tal senso, anche se entrambi gli account fanno parte della

stessa organizzazione AWS o della stessa unità organizzativa. Puoi utilizzare le unità organizzative per raggruppare gli account che desideri amministrare come una singola unità.

Il passaggio successivo consiste nel ridurre il set massimo di autorizzazioni che è possibile concedere ai principali all'interno degli account dei membri dell'organizzazione. A tale scopo, puoi utilizzare le policy di controllo dei servizi, applicabili a un'unità organizzativa o a un account. Le policy di controllo dei servizi possono applicare controlli di accesso comuni, ad esempio limitare l'accesso a Regioni AWS specifiche, aiutare a prevenire l'eliminazione di risorse o disabilitare azioni di servizio potenzialmente rischiose. Le policy di controllo dei servizi applicate alla radice dell'organizzazione influiscono solo sugli account dei membri, non sull'account di gestione. Le policy di controllo dei servizi regolano solo i principali all'interno della tua organizzazione. Le tue policy di controllo dei servizi non regolano i principali esterni alla tua organizzazione che accedono alle tue risorse.

Se utilizzi AWS Control Tower, puoi sfruttare i controlli e le zone di destinazione come base per i guardrail delle autorizzazioni e l'ambiente multi-account. Le zone di destinazione forniscono un ambiente di base preconfigurato e sicuro, con account separati per diversi carichi di lavoro e applicazioni. I guardrail impongono controlli obbligatori su sicurezza, operazioni e conformità attraverso una combinazione di policy di controllo dei servizi, regole AWS Config e altre configurazioni. Tuttavia, quando si utilizzano guardrail e zone di destinazione di Control Tower insieme a SCP personalizzati dell'organizzazione, è fondamentale seguire le best practice descritte nella documentazione AWS per evitare conflitti e garantire una governance adeguata. Per suggerimenti dettagliati sulla gestione di SCP, account e unità organizzative (UO) in un ambiente Control Tower, fai riferimento alla guida di AWS Control Tower per AWS Organizations.

Se ti attieni a queste linee guida, puoi sfruttare efficacemente i guardrail, le zone di destinazione e gli SCP personalizzati di Control Tower, riducendo al contempo i potenziali conflitti e garantendo una governance e un controllo adeguati sull'ambiente AWS multi-account.

Un ulteriore passo consiste nell'utilizzare le policy delle risorse IAM per definire le azioni disponibili che puoi intraprendere sulle risorse da esse governate, oltre a tutte le condizioni che il principale che agisce deve soddisfare. Questo può essere un ambito ampio, come consentire tutte le azioni fintanto che il principale fa parte dell'organizzazione (utilizzando la chiave di condizione PrincipalOrgId), o granulare, come consentire solo azioni specifiche da parte di un ruolo IAM specifico. Puoi adottare un approccio simile con le condizioni nelle policy di attendibilità del ruolo IAM. Se una policy di attendibilità di una risorsa o di un ruolo nomina esplicitamente un principale nello stesso account del ruolo o della risorsa che governa, tale principale non ha bisogno di una policy IAM associata che conceda le stesse autorizzazioni. Se il principale si trova in un account diverso dalla risorsa, deve disporre di una policy IAM associata che conceda tali autorizzazioni.

Spesso, un team addetto al carico di lavoro vorrà gestire le autorizzazioni richieste dal proprio carico di lavoro. Ciò potrebbe richiedere al team di creare nuovi ruoli IAM e policy di autorizzazione. Puoi definire l'ambito massimo di autorizzazioni che il team può concedere in un limite delle autorizzazioni IAM e associare questo documento a un ruolo IAM, utilizzabile dal team per gestire autorizzazioni e ruoli IAM. Questo approccio può fornire la flessibilità necessaria per completare il lavoro, mitigando al contempo i rischi legati all'accesso amministrativo IAM.

Un passaggio più granulare consiste nell'implementazione delle tecniche di gestione degli accessi privilegiati (PAM) e di gestione temporanea degli accessi elevati (TEAM). Un esempio di gestione degli accessi privilegiati consiste nel richiedere ai principali di eseguire l'autenticazione a più fattori prima di intraprendere azioni privilegiate. Per ulteriori informazioni, consulta Configuring MFA-protected API access. La gestione temporanea degli accessi elevati richiede una soluzione che gestisca l'approvazione e i tempi in cui un principale può avere un accesso elevato. Un approccio consiste nell'aggiungere temporaneamente il principale alla policy di attendibilità dei ruoli per un ruolo IAM con accesso elevato. Un altro approccio consiste nel ridurre, in condizioni di funzionamento normale, le autorizzazioni concesse a un principale da un ruolo IAM mediante una policy di sessione, quindi revocare in modo temporaneo questa restrizione durante la finestra temporale approvata. Per ulteriori informazioni sulle soluzioni convalidate da AWS e da alcuni partner selezionati, consulta Temporary elevated access.

Passaggi dell'implementazione

1. Isola i carichi di lavoro e gli ambienti in Account AWS separati.
2. Usa le policy di controllo del servizio per ridurre il set massimo di autorizzazioni che possono essere concesse ai principali all'interno degli account membri della tua organizzazione.
    a. Quando si definiscono SCP per ridurre l'insieme massimo di autorizzazioni che possono essere concesse ai principali all'interno degli account membri dell'organizzazione, è possibile scegliere tra un approccio di tipo elenco di consentiti o elenco di rifiuto. La strategia dell'elenco di consentiti specifica esplicitamente gli accessi consentiti e blocca implicitamente tutti gli altri accessi. La strategia dell'elenco di rifiuto specifica esplicitamente gli accessi non consentiti e consente tutti gli altri accessi per impostazione predefinita. Entrambe le strategie presentano vantaggi e compromessi e la scelta appropriata dipende dai requisiti specifici e dal modello di rischio dell'organizzazione. Per maggiori dettagli, consulta Strategy for using SCPs.
    b. Inoltre, esamina gli esempi di policy di controllo dei servizi per capire come creare le SCP in modo efficace.

3. Utilizza le policy relative alle risorse IAM per definire l'ambito e specificare le condizioni per le azioni consentite sulle risorse. Utilizza le condizioni nelle policy di fiducia dei ruoli IAM per creare restrizioni all'assunzione dei ruoli.
4. Assegna limiti delle autorizzazioni IAM ai ruoli IAM che i team del carico di lavoro possono quindi utilizzare per autorizzazioni e ruoli IAM del proprio carico di lavoro.
5. Valuta le soluzioni PAM e TEAM in base alle tue esigenze.

Risorse

Documenti correlati:

- Data perimeters on AWS
- Establish permissions guardrails using data perimeters
- Logica di valutazione delle policy

Esempi correlati:

- Service control policy examples

Strumenti correlati:

- AWS Solution: Temporary Elevated Access Management
- Validated security partner solutions for TEAM

SEC03-BP06 Gestione degli accessi in base al ciclo di vita

Monitora e regola le autorizzazioni concesse ai tuoi principali (utenti, ruoli e gruppi) durante il loro ciclo di vita all'interno dell'organizzazione. Adatta le appartenenze ai gruppi quando gli utenti cambiano ruolo e rimuovi l'accesso quando un utente lascia l'organizzazione.

Risultato desiderato: monitori e modifichi le autorizzazioni durante l'intero ciclo di vita dei principali all'interno dell'organizzazione, riducendo così il rischio di privilegi superflui. Concedi l'accesso appropriato quando crei un utente. L'accesso viene modificato man mano che cambiano le responsabilità dell'utente e lo si rimuove quando l'utente non è più attivo o ha lasciato l'organizzazione. Gestisci a livello centrale le modifiche ai tuoi utenti, ruoli e gruppi. Utilizza l'automazione per propagare le modifiche agli ambienti AWS.

Anti-pattern comuni:

- Concedere in anticipo alle identità privilegi di accesso eccessivi o ampi, al di là di quanto richiesto inizialmente.
- Non rivedere né modificare i privilegi di accesso in base al cambiamento dei ruoli e delle responsabilità delle identità nel tempo.
- Lasciare le identità inattive o terminate con privilegi di accesso attivi. Ciò aumenta il rischio di accessi non autorizzati.
- Non sfruttare l'automazione per gestire il ciclo di vita delle identità.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Gestisci e adatta attentamente i privilegi di accesso che concedi alle identità (come utenti, ruoli, gruppi) durante il loro ciclo di vita. Questo ciclo di vita include la fase iniziale di onboarding, i continui cambiamenti di ruoli e responsabilità e l'eventuale offboarding o cessazione. Gestisci in modo proattivo l'accesso in base alla fase del ciclo di vita per mantenere il livello di accesso appropriato. Rispetta il principio del privilegio minimo per ridurre il rischio di privilegi di accesso eccessivi o non necessari.

Puoi gestire il ciclo di vita degli utenti IAM direttamente all'interno dell'Account AWS o tramite federazione dal gestore dell'identità digitale della forza lavoro al Centro identità AWS IAM. Per gli utenti IAM, puoi creare, modificare ed eliminare gli utenti e le relative autorizzazioni associate nell'Account AWS. Per gli utenti federati, puoi utilizzare il Centro identità IAM per gestire il ciclo di vita sincronizzando le informazioni sugli utenti e sui gruppi dal gestore dell'identità digitale dell'organizzazione mediante il protocollo System for Cross-domain Identity Management (SCIM).

SCIM è un protocollo standard aperto per il provisioning e il deprovisioning automatici delle identità degli utenti su diversi sistemi. Integrando il tuo gestore dell'identità digitale con il Centro identità IAM tramite SCIM, puoi sincronizzare in automatico le informazioni sugli utenti e sui gruppi, verificando che i privilegi di accesso siano concessi, modificati o revocati in base ai cambiamenti nella fonte di identità autorevole dell'organizzazione.

Man mano che i ruoli e le responsabilità dei dipendenti cambiano all'interno dell'organizzazione, modifica di conseguenza i loro privilegi di accesso. Puoi utilizzare i set di autorizzazioni del Centro identità IAM per definire diversi ruoli o responsabilità lavorative e associarli alle policy IAM e alle autorizzazioni appropriate. Quando il ruolo di un dipendente cambia, puoi aggiornare il set di

autorizzazioni assegnato per riflettere le nuove responsabilità. Verifica che il dipendente disponga dell'accesso necessario rispettando il principio del privilegio minimo.

Passaggi dell'implementazione

1. Definisci e documenta un processo del ciclo di vita della gestione degli accessi, comprese le procedure per la concessione dell'accesso iniziale, le revisioni periodiche e l'offboarding.
2. Implementa ruoli IAM, gruppi e limiti delle autorizzazioni per gestire l'accesso collettivamente e applicare i livelli di accesso massimi consentiti.
3. Effettua l'integrazione con un gestore dell'identità digitale federato (come Microsoft Active Directory, Okta, Ping Identity) come fonte autorevole per le informazioni sugli utenti e sui gruppi utilizzando il Centro identità IAM.
4. Utilizza il protocollo SCIM per sincronizzare le informazioni su utenti e gruppi dal gestore dell'identità digitale nell'Identity Store del Centro identità IAM.
5. Crea set di autorizzazioni nel Centro identità IAM che rappresentino diversi ruoli o responsabilità all'interno dell'organizzazione. Definisci autorizzazioni e policy IAM appropriate per ogni set di autorizzazioni.
6. Implementa revisioni regolari degli accessi, la relativa revoca tempestiva e il miglioramento continuo del processo del ciclo di vita della gestione degli accessi.
7. Offri formazione e sensibilizza i dipendenti in materia di best practice sulla gestione degli accessi.

Risorse

Best practice correlate:

- SEC02-BP04 Fai affidamento su un gestore dell'identità digitale centralizzato

Documenti correlati:

- Manage your identity source
- Manage identities in IAM Identity Center
- Uso di AWS Identity and Access Management Access Analyzer
- IAM Access Analyzer policy generation

Video correlati:

- AWS re:Inforce 2023 - Manage temporary elevated access with AWS IAM Identity Center
- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center
- AWS re:Invent 2022 - Harness power of IAM policies & rein in permissions w/Access Analyzer

SEC03-BP07 Analisi dell'accesso multi-account e pubblico

Monitora continuamente i risultati che evidenziano l'accesso multi-account e pubblico. Limita l'accesso multi-account e pubblico alle risorse che lo richiedono.

Risultato desiderato: conosci le risorse AWS condivise e con chi avviene la condivisione. Monitora e sottoponi costantemente ad audit le risorse condivise per verificare che siano condivise solo con i principali autorizzati.

Anti-pattern comuni:

- Assenza di un inventario delle risorse condivise.
- Mancanza di un processo di approvazione dell'accesso multi-account e dell'accesso pubblico alle risorse.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Se l'account è in AWS Organizations, puoi concedere l'accesso alle risorse all'intera organizzazione, a specifiche unità organizzative o a singoli account. Se l'account non è membro di un'organizzazione, puoi condividere le risorse con account individuali. Puoi concedere l'accesso multi-account diretto utilizzando policy basate sulle risorse, ad esempio le policy di bucket di Amazon Simple Storage Service (Amazon S3) o consentendo a un principale di un altro account di assumere un ruolo IAM nel tuo account. Quando utilizzi le policy sulle risorse, verifica che l'accesso sia concesso solo ai principali autorizzati. Definisci un processo per approvare tutte le risorse che devono essere pubblicamente disponibili.

AWS Identity and Access Management Access Analyzer utilizza una sicurezza comprovabile per identificare tutti i percorsi di accesso a una risorsa dall'esterno del proprio account. Esamina continuamente le policy delle risorse e segnala i risultati dell'accesso multi-account e pubblico per semplificare l'analisi di accessi potenzialmente estesi. Prendi in considerazione la configurazione di IAM Access Analyzer con AWS Organizations per verificare di avere visibilità su tutti i tuoi account. IAM Access Analyzer consente inoltre di visualizzare in anteprima i risultati prima di implementare

le autorizzazioni per le risorse. In questo modo è possibile verificare che le modifiche alle policy garantiscano alle risorse solo l'accesso multi-account e pubblico previsto. In caso di progettazione per l'accesso multi-account, puoi utilizzare policy di affidabilità per controllare i casi in cui è possibile assumere un ruolo. Ad esempio, puoi utilizzare la chiave di condizione `PrincipalOrgId` per negare un tentativo di assumere un ruolo al di fuori di AWS Organizations.

AWS Config è grado di segnalare le risorse non configurate correttamente. Inoltre, tramite i controlli delle policy AWS Config, rileva le risorse con l'accesso pubblico configurato. Servizi come AWS Control Tower e AWS Security Hub CSPM semplificano l'implementazione di controlli di rilevamento e guardrail in AWS Organizations per identificare e correggere le risorse pubblicamente esposte. Ad esempio, AWS Control Tower dispone di un guardrail gestito in grado di rilevare se eventuali snapshot Amazon EBS sono ripristinabili tramite Account AWS.

Passaggi dell'implementazione

- Prendi in considerazione l'utilizzo di AWS Config per AWS Organizations: AWS Config consente di aggregare gli esiti di più account all'interno di AWS Organizations in un account amministratore delegato. In questo modo, avrai una visione completa e potrai eseguire l'implementazione di Regole di AWS Config su più account per rilevare risorse accessibili al pubblico.
- Configurare AWS Identity and Access Management Access Analyzer: consente di identificare le risorse nell'organizzazione e negli account, ad esempio bucket Amazon S3 o ruoli IAM, condivise con un'entità esterna.
- Usa la riparazione automatica in AWS Config per rispondere alle modifiche nella configurazione dell'accesso pubblico dei bucket Amazon S3: puoi attivare in automatico le impostazioni di blocco dell'accesso pubblico per i bucket Amazon S3.
- Implementa monitoraggio e avvisi per stabilire se i bucket Amazon S3 sono diventati pubblici: devi disporre di monitoraggio e avvisi per stabilire se il blocco dell'accesso pubblico Amazon S3 è disattivato e se i bucket Amazon S3 diventano pubblici. Inoltre, se utilizzi AWS Organizations, puoi creare una policy di controllo dei servizi che impedisca modifiche alle policy di accesso pubblico di Amazon S3. AWS Trusted Advisor verifica la presenza di bucket Amazon S3 dotati di autorizzazioni di accesso aperte. Le autorizzazioni bucket che concedono, caricano o eliminano l'accesso per chiunque danno origine a potenziali problemi di sicurezza, consentendo a chiunque di aggiungere, modificare o rimuovere elementi in un bucket. Il controllo di Trusted Advisor esamina le autorizzazioni bucket esplicite e le policy associate che possono prevalere sulle autorizzazioni bucket. Puoi anche utilizzare AWS Config per monitorare l'accesso pubblico ai bucket Amazon S3. Per ulteriori informazioni, consulta How to Use AWS Config to Monitor for and Respond to Amazon S3 Buckets Allowing Public Access.

Quando si esaminano i controlli di accesso per i bucket Amazon S3, è importante considerare la natura dei dati memorizzati al loro interno. Amazon Macie è un servizio progettato per aiutarti a scoprire e proteggere dati sensibili, come informazioni di identificazione personale (PII), dati sanitari protetti (PHI) e credenziali come chiavi private o chiavi di accesso AWS.

Risorse

Documenti correlati:

- Using AWS Identity and Access Management Access Analyzer
- AWS Control Tower controls library
- AWS Foundational Security Best Practices standard
- AWS Config Regole gestite da
- AWS Trusted Advisor check reference
- Monitoring AWS Trusted Advisor check results with Amazon EventBridge
- Managing AWS Config Rules Across All Accounts in Your Organization
- AWS Config e AWS Organizations
- Make your AMI publicly available for use in Amazon EC2

Video correlati:

- Best Practices for securing your multi-account environment
- Dive Deep into IAM Access Analyzer

SEC03-BP08 Condivisione delle risorse in modo sicuro all'interno dell'organizzazione

Con l'aumento del numero di carichi di lavoro, è possibile che sia necessario condividere l'accesso alle risorse in tali carichi di lavoro o eseguire il provisioning delle risorse più volte su più account. Possono esistere costrutti per segmentare il proprio l'ambiente, come ambienti di sviluppo, di test e di produzione. Tuttavia, la presenza di costrutti di separazione non limita la possibilità di condivisione sicura. La condivisione di componenti sovrapposti consente di ridurre i costi operativi e di garantire un'esperienza coerente, senza dover intuire cosa potrebbe sfuggire durante la creazione della stessa risorsa più volte.

Risultato desiderato: ridurre al minimo gli accessi involontari tramite l'uso di metodi sicuri di condivisione delle risorse all'interno dell'organizzazione e contribuire alle iniziative di prevenzione della perdita dei dati. Ridurre i costi operativi rispetto alla gestione dei singoli componenti, ridurre gli

errori dovuti alla creazione manuale dello stesso componente più volte e aumentare la scalabilità dei carichi di lavoro. Si riducono i tempi di risoluzione in caso di guasti multipli e si aumenta la sicurezza nel determinare quando un componente non è più necessario. Per linee guida prescrittive sull'analisi delle risorse condivise all'esterno, consulta SEC03-BP07 Analisi dell'accesso multi-account e pubblico.

Anti-pattern comuni:

- Mancanza di un processo per il monitoraggio continuo e segnalazione automatica di condivisioni esterne inaspettate.
- Mancanza di una linea di base su ciò che deve e ciò che non deve essere condiviso.
- Scelta di una policy di ampia apertura piuttosto che di una condivisione esplicita quando richiesto.
- Creazione manuale di risorse fondamentali che si sovrappongono quando necessario.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Progetta controlli e modelli di accesso per gestire il consumo di risorse condivise in modo sicuro e solo con entità fidate. Monitora le risorse condivise e controllane in modo costante l'accesso, ricevendo un avviso in caso di condivisione inappropriata o inaspettata. Consulta Analisi dell'accesso multi-account e pubblico per stabilire una governance che riduca l'accesso esterno alle sole risorse che lo richiedono e per stabilire un processo di monitoraggio continuo e avvisi automatici.

La condivisione tra più account all'interno di AWS Organizations è supportata da diversi servizi AWS, come AWS Security Hub CSPM, Amazon GuardDuty e AWS Backup. Questi servizi permettono di condividere i dati con un account centrale, di accedere a un account centrale o di gestire risorse e dati da un account centrale. Ad esempio, AWS Security Hub CSPM può trasferire gli esiti dai singoli account a un account centrale in cui è possibile visualizzare tutti gli esiti. AWS Backup può eseguire un backup di una risorsa e condividerlo tra gli account. Puoi usare AWS Resource Access Manager (AWS RAM) per la condivisione di altre risorse comuni, come sottoreti VPC e collegamenti del gateway di transito alla VPN, AWS Network Firewall o pipeline IA di Amazon SageMaker.

Per limitare l'account in modo che condivida sole risorse all'interno dell'organizzazione, utilizza le policy di controllo dei servizi (SCP) per impedire l'accesso a principali esterni. In caso di condivisione di risorse, combina controlli basati sull'identità e di rete per creare un perimetro di dati per l'organizzazione e proteggere la stessa da accessi involontari. Un perimetro di dati è un insieme di guardrail preventivi che aiutano a verificare che solo le identità fidate accedano a

risorse fidate dalle reti previste. Questi controlli pongono limiti adeguati alle risorse condivisibili e impediscono la condivisione o l'esposizione di risorse che non sono consentite. Ad esempio, nell'ambito del tuo perimetro di dati, puoi utilizzare le policy degli endpoint VPC e la condizione `AWS:PrincipalOrgId` per garantire che le identità che accedono ai bucket Amazon S3 appartengano alla tua organizzazione. È importante sottolineare che le SCP non si applicano a ruoli collegati a servizi o a principali dei servizi AWS.

Se usi Amazon S3, disattiva gli ACL per il bucket Amazon S3 e definisci il controllo degli accessi con le policy IAM. Per limitare l'accesso a un'origine Amazon S3 da Amazon CloudFront, effettua la migrazione dall'identità di accesso origine (OAI) al controllo di accesso origine (OAC) che supporta funzionalità aggiuntive, tra cui la crittografia lato server con AWS Key Management Service.

In alcuni casi, può essere necessario condividere le risorse al di fuori dell'organizzazione o concedere a terze parti l'accesso alle risorse stesse. Per linee guida prescrittive sulla gestione delle autorizzazioni per la condivisione esterna delle risorse, consulta Gestione delle autorizzazioni.

Passaggi dell'implementazione

1. Utilizzo di AWS Organizations - AWS Organizations è un servizio di gestione degli account che consente di consolidare più Account AWS in un'organizzazione, che è possibile creare e gestire in modo centralizzato. È possibile raggruppare gli account in unità organizzative (OU) e associare policy diverse a ciascuna di esse per soddisfare le esigenze di bilancio, sicurezza e conformità. È inoltre possibile controllare il modo in cui i servizi di Intelligenza Artificiale (IA) e di machine learning (ML) di AWS possono raccogliere e archiviare i dati e utilizzare la gestione multi-account dei servizi AWS integrati nelle organizzazioni.
2. Integrazione di AWS Organizations con i servizi AWS - Se usi un servizio AWS per eseguire attività per tuo conto negli account membri dell'organizzazione, AWS Organizations crea un ruolo collegato ai servizi IAM (SLR) per tale servizio in ogni account membro. L'accesso attendibile deve essere gestito tramite la Console di gestione AWS, le API AWS o la AWS CLI. Per linee guida prescrittive sull'attivazione dell'accesso attendibile, consulta Using AWS Organizations with other AWS services e AWS services that you can use with Organizations.
3. Definizione di un perimetro dati - Un perimetro dati fornisce un limite ben chiaro per attendibilità e proprietà. Su AWS, solitamente è rappresentato come la tua organizzazione AWS gestita tramite AWS Organizations, insieme a eventuali sistemi o reti on-premises che accedono alle tue risorse AWS. L'obiettivo del perimetro dati è verificare che l'accesso sia consentito se l'identità è attendibile, la risorsa è attendibile e la rete è conforme. Tuttavia, la definizione di un perimetro di dati non è una soluzione adatta a tutti gli scenari. Valuta e adotta gli obiettivi di controllo delineati nel white paper Building a Perimeter on AWS in base ai tuoi specifici modelli e requisiti di rischio

per la sicurezza. Devi valutare attentamente la tua specifica posizione di rischio e implementare i controlli perimetrali in linea con le tue esigenze di sicurezza.

4. Utilizzo della condivisione delle risorse nei servizi AWS e restrizioni correlate - Molti servizi AWS consentono di condividere risorse con altri account o di destinare risorse ad altri account, come ad esempio Amazon Machine Image (AMI) e AWS Resource Access Manager (AWS RAM). Limita l'API `ModifyImageAttribute` in modo da specificare gli account affidabili con cui condividere l'AMI. Specifica la condizione `ram:RequestedAllowsExternalPrincipals` in caso di utilizzo di AWS RAM per limitare la condivisione solo alla tua organizzazione e impedire l'accesso di identità non attendibili. Per considerazioni e linee guida prescrittive, consulta Resource sharing and external targets.
5. Utilizzo di AWS RAM per condividere risorse in modo sicuro all'interno di un account o con altri Account AWS - AWS RAM ti consente di condividere in modo sicuro le risorse create con i ruoli e gli utenti del tuo account e di altri Account AWS. In un ambiente multi-account, AWS RAM consente di creare una risorsa una sola volta e di condividerla con altri account. Questo approccio contribuisce a ridurre i costi operativi, fornendo al contempo coerenza, visibilità e la facilità di audit grazie alle integrazioni con Amazon CloudWatch e AWS CloudTrail, che non si ottengono quando si utilizza l'accesso multi-account.

   Se disponi di risorse condivise in precedenza mediante una policy basata sulle risorse, puoi utilizzare l'API `PromoteResourceShareCreatedFromPolicy` o un metodo equivalente per promuovere il passaggio da una condivisione di risorse a una condivisione completa di risorse completa AWS RAM.

   In alcuni casi, potrebbe essere necessario adottare ulteriori misure per condividere le risorse. Ad esempio, per condividere uno snapshot crittografato, occorre condividere una chiave AWS KMS.

Risorse

Best practice correlate:

- SEC03-BP07 Analisi dell’accesso multi-account e pubblico
- SEC03-BP09 Condivisione sicura delle risorse con terze parti
- SEC05-BP01 Creazione di livelli di rete

Documenti correlati:

- Bucket owner granting cross-account permission to objects it does not own

- How to use Trust Policies with IAM
- Building Data Perimeter on AWS
- Come utilizzare un ID esterno quando si concede a una terza parte l'accesso alle risorse AWS
- AWS services you can use with AWS Organizations
- Establishing a data perimeter on AWS: Allow only trusted identities to access company data

Video correlati:

- Granular Access with AWS Resource Access Manager
- Securing your data perimeter with VPC endpoints
- Establishing a data perimeter on AWS

Strumenti correlati:

- Esempi di policy del perimetro di dati

SEC03-BP09 Condivisione sicura delle risorse con terze parti

La sicurezza dell'ambiente cloud non si ferma alla tua organizzazione. L'organizzazione potrebbe affidare a terze parti la gestione di una parte dei dati. La gestione dei permessi per il sistema gestito da terze parti deve seguire la pratica dell'accesso just-in-time utilizzando il principio del privilegio minimo con credenziali temporanee. Lavorando a stretto contatto con una terza parte, puoi ridurre allo stesso momento la portata dell'impatto e il rischio di accesso non intenzionale.

Risultato desiderato: non vengono utilizzate credenziali AWS Identity and Access Management (IAM) a lungo termine come chiavi di accesso e chiavi segrete, poiché rappresentano un rischio per la sicurezza se utilizzate in modo improprio. Al contrario, vengono utilizzati i ruoli IAM e le credenziali temporanee per migliorare il livello di sicurezza e ridurre al minimo il sovraccarico operativo legato alla gestione delle credenziali a lungo termine. Quando concedi l'accesso a terze parti, utilizzi un identificativo univoco universale (UUID) come ID esterno nella policy di attendibilità IAM e mantieni sotto il tuo controllo le policy IAM collegate al ruolo per garantire l'accesso con il privilegio minimo. Per linee guida prescrittive sull'analisi delle risorse condivise a livello esterno, consulta SEC03-BP07 Analisi dell'accesso multi-account e pubblico.

Anti-pattern comuni:

- Utilizzo della policy di attendibilità IAM predefinita senza alcuna condizione.

- Utilizzo di credenziali IAM e chiavi di accesso a lungo termine.
- Riutilizzo di ID esterni.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

In alcuni casi, può essere necessario condividere le risorse al di fuori di AWS Organizations o concedere a terze parti l'accesso alle risorse stesse. Ad esempio, una terza parte potrebbe fornire una soluzione di monitoraggio che necessita di accedere alle risorse del tuo account. In questi casi, devi creare un ruolo IAM multi-account con i soli privilegi necessari alla terza parte. Inoltre, definisci una policy di attendibilità utilizzando la condizione ID esterno. L'utilizzo di un ID esterno da parte tua o della terza parte può comportare la generazione di un ID univoco per ogni cliente, terza parte o tenancy. Una volta creato, l'ID univoco non deve essere controllato da nessuno, se non da te. La terza parte deve implementare un processo per collegare l'ID esterno al cliente in modo sicuro, verificabile e riproducibile.

Puoi anche utilizzare IAM Roles Anywhere per la gestione dei ruoli IAM per le applicazioni all'esterno di AWS che non utilizzano API AWS.

Se la terza parte non ha più bisogno di accedere al tuo ambiente, rimuovi il ruolo. Evita di fornire a terze parti credenziali a lungo termine. Mantieni la visibilità degli altri servizi AWS che supportano la condivisione, come AWS Well-Architected Tool che consente di condividere un carico di lavoro con altri Account AWS e AWS Resource Access Manager che permette di condividere in modo sicuro una risorsa AWS di tua proprietà con altri account.

Passaggi dell'implementazione

1. Utilizza ruoli multi-account per fornire l'accesso agli account esterni. I ruoli multi-account riducono la quantità di informazioni sensibili archiviate da account esterni e terze parti per l'assistenza ai propri clienti. I ruoli multi-account consentono di concedere l'accesso alle risorse AWS dell'account in modo sicuro a terze parti, come i Partner AWS o altri account dell'organizzazione, mantenendo la possibilità di gestire e sottoporre a audit tale accesso. La terza parte può fornire il servizio da un'infrastruttura ibrida o, in alternativa, estrarre i dati in una sede esterna. IAM Roles Anywhere consente ai carichi di lavoro di terze parti di interagire in modo sicuro con i tuoi carichi di lavoro AWS e di ridurre ulteriormente la necessità di credenziali a lungo termine.

Non devi utilizzare credenziali a lungo termine o chiavi di accesso associate agli utenti per fornire accesso ad account esterni. Per fornire l'accesso multi-account invece, occorre utilizzare i ruoli multi-account.

2. Effettua verifiche di due diligence e garantisci un accesso sicuro per i provider SaaS di terze parti. Quando condividi alcune risorse con provider SaaS di terze parti, esegui un'attenta due diligence per assicurarti che abbiano un approccio sicuro e responsabile all'accesso alle tue risorse AWS. Valuta il loro modello di responsabilità condivisa per capire quali misure di sicurezza forniscono e cosa rientra nella tua responsabilità. Assicurati che il provider SaaS disponga di un processo sicuro e verificabile per l'accesso alle tue risorse, incluso l'uso di ID esterni e principi di accesso con privilegio minimo. L'uso di ID esterni aiuta a risolvere i problemi di "confused deputy".

   Implementa controlli di sicurezza per garantire un accesso sicuro e il rispetto del principio del privilegio minimo quando concedi l'accesso a provider SaaS di terze parti. Ciò può includere l'uso di ID esterni, di identificatori univoci universali (UUID) e di policy di attendibilità IAM che limitano l'accesso solo a ciò che è strettamente necessario. Collabora a stretto contatto con il provider SaaS per stabilire meccanismi di accesso sicuri, controllarne regolarmente l'accesso alle risorse AWS e condurre audit per garantire il rispetto dei requisiti di sicurezza.
3. Rendi obsolete le credenziali a lungo termine fornite dal cliente. Rendi obsoleto l'uso di credenziali a lungo termine e utilizza ruoli multi-account oppure IAM Roles Anywhere. Se devi utilizzare credenziali a lungo termine, stabilisci un piano per migrare verso l'accesso basato sui ruoli. Per i dettagli sulla gestione delle chiavi, consulta Gestione delle identità. Collabora inoltre con il team dell'Account AWS e con la terza parte per predisporre un runbook di mitigazione dei rischi. Per un prontuario su come rispondere e mitigare il potenziale impatto di un incidente di sicurezza, consulta Risposta agli imprevisti.
4. Verifica che la configurazione presenti indicazioni prescrittive o sia automatizzata. L'ID esterno non viene trattato come un segreto, ma non deve essere un valore facilmente individuabile, come un numero di telefono, un nome o un ID account. Rendi l'ID esterno un campo di sola lettura, in modo che non possa essere modificato per rappresentare la configurazione.

   L'ID esterno può essere generato da te o dalla terza parte. Definisci un processo per stabilire chi è responsabile della generazione dell'ID. Indipendentemente dall'entità che crea l'ID esterno, la terza parte fa rispettare l'univocità e i formati in modo coerente tra i clienti.

   La policy creata per l'accesso multi-account ai tuoi account deve attenersi al principio del privilegio minimo. La terza parte deve fornire un documento sulla policy del ruolo o un meccanismo di configurazione automatica che utilizzi un modello AWS CloudFormation o un equivalente

per l'utente. In questo modo si riduce la possibilità di errori associati alla creazione manuale della policy e si offre un audit trail. Per ulteriori informazioni sull'utilizzo di un modello AWS CloudFormation per creare ruoli multi-account, consulta Cross-Account Roles.

La terza parte deve fornire un meccanismo di configurazione automatizzato e verificabile. Tuttavia, utilizzando il documento della policy sui ruoli che delinea gli accessi necessari, è possibile automatizzare l'impostazione del ruolo. Con un modello AWS CloudFormation o equivalente, è necessario monitorare le modifiche con il rilevamento delle deviazioni come parte della pratica di audit.

5. Tieni conto delle modifiche. La struttura del tuo account, la tua necessità di una terza parte o l'offerta di servizi che ti viene fornita possono cambiare. Occorre anticipare cambiamenti e guasti, quindi pianificare di conseguenza con le persone, i processi e le tecnologie adeguati. Sottoponi periodicamente a audit il livello di accesso fornito e implementa metodi di rilevamento per avvisare l'utente di cambiamenti inattesi. Monitora e sottoponi ad audit l'uso del ruolo e del datastore degli ID esterni. Occorre essere pronti a revocare l'accesso a terze parti, in modo temporaneo o permanente, in seguito a modifiche o modelli di accesso imprevisti. Inoltre, valuta l'impatto dell'operazione di revoca, compreso il tempo necessario per eseguirla, le persone coinvolte, il costo e l'impatto su altre risorse.

   Per linee guida prescrittive sui metodi di rilevamento, consulta le best practice di rilevamento.

Risorse

Best practice correlate:

- SEC02-BP02 Utilizzo di credenziali temporanee
- SEC03-BP05 Definizione dei guardrail per le autorizzazioni dell'organizzazione
- SEC03-BP06 Gestione degli accessi in base al ciclo di vita
- SEC03-BP07 Analisi dell'accesso multi-account e pubblico
- SEC04 Rilevamento

Documenti correlati:

- Bucket owner granting cross-account permission to objects it does not own
- How to use trust policies with IAM roles
- Delegate access across Account AWS using IAM roles

- How do I access resources in another Account AWS using IAM?
- Best practice per la sicurezza in IAM
- Cross-account policy evaluation logic
- How to use an external ID when granting access to your AWS resources to a third party
- Collecting Information from AWS CloudFormation Resources Created in External Accounts with Custom Resources
- Securely Using External ID for Accessing AWS Accounts Owned by Others
- Extend IAM roles to workloads outside of IAM with IAM Roles Anywhere

Video correlati:

- How do I allow users or roles in a separate Account AWS access to my Account AWS?
- AWS re:Invent 2018: Become an IAM Policy Master in 60 Minutes or Less
- AWS Knowledge Center Live: IAM Best Practices and Design Decisions

Esempi correlati:

- Configurazione dell'accesso multi-account in Amazon DynamoDB
- Strumento di query di rete AWS STS

Rilevamento

Domanda

- SEC 4. In che modo individui ed esamini gli eventi di sicurezza?

SEC 4. In che modo individui ed esamini gli eventi di sicurezza?

Acquisisci e analizza gli eventi a partire da log e metriche per acquistare visibilità. Agisci su eventi di sicurezza e potenziali minacce per contribuire a rendere sicuro il carico di lavoro.

Best practice

- SEC04-BP01 Configurazione dei log di servizi e applicazioni
- SEC04-BP02 Acquisizione di log, esiti e metriche in posizioni standardizzate
- SEC04-BP03 Correlazione e arricchimento degli avvisi di sicurezza

- SEC04-BP04 Avvio della riparazione delle risorse non conformi

SEC04-BP01 Configurazione dei log di servizi e applicazioni

Mantieni i log degli eventi di sicurezza dei servizi e delle applicazioni. Si tratta di un principio fondamentale di sicurezza per i casi d'uso di audit, indagini e operazioni, nonché di un requisito di sicurezza comune guidato da standard, policy e procedure di governance, rischio e conformità (GRC).

Risultato desiderato: un'organizzazione deve essere in grado di recuperare in modo affidabile e coerente i log degli eventi di sicurezza da servizi e applicazioni AWS in modo tempestivo, laddove necessario, per adempiere a un processo o obbligo interno, come la risposta a un incidente di sicurezza. Considera la possibilità di centralizzare i log per migliori risultati operativi.

Anti-pattern comuni:

- Log archiviati in modo perpetuo o eliminati troppo presto.
- Tutti possono accedere ai log.
- Affidamento totale a processi manuali per la governance e l'utilizzo dei log.
- Archiviazione di ogni singolo tipo di log nel caso in cui sia necessario.
- Controllo dell'integrità del log solo quando è necessario.

Vantaggi dell'adozione di questa best practice: implementare un meccanismo di analisi della causa principale (RCA) per gli incidenti di sicurezza e una fonte di prove per gli obblighi in termini di governance, rischio e conformità.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Durante un'indagine di sicurezza o in altri casi d'uso basati sui tuoi requisiti, devi essere in grado di esaminare i log pertinenti per registrare e comprendere l'intera portata e la tempistica dell'incidente. I log sono necessari anche per la generazione di avvisi, che indicano il verificarsi di determinate azioni di interesse. È fondamentale selezionare, attivare, memorizzare e configurare i meccanismi di query e recupero e gli avvisi.

Passaggi dell'implementazione

- Seleziona e utilizza le origini dei log. Prima di un'indagine di sicurezza, devi acquisire i log pertinenti per ricostruire in modo retroattivo l'attività in un Account AWS. Seleziona le origini dei log pertinenti per i carichi di lavoro.

  I criteri di selezione delle origini dei log devono basarsi sui casi d'uso richiesti dall'azienda. Stabilisci un percorso per ogni Account AWS utilizzando AWS CloudTrail o un percorso AWS Organizations e configura per lo stesso un bucket Amazon S3.

  AWS CloudTrail è un servizio di creazione di log che tiene traccia delle chiamate API effettuate su un Account AWS, acquisendo l'attività del servizio AWS. È attivato per impostazione predefinita, con una conservazione di 90 giorni degli eventi di gestione, recuperabili tramite la cronologia degli eventi di CloudTrail mediante Console di gestione AWS, AWS CLI o un SDK AWS. Per una maggiore conservazione e visibilità degli eventi relativi ai dati, crea un percorso CloudTrail e associalo a un bucket Amazon S3 e, facoltativamente, a un gruppo di log Amazon CloudWatch. In alternativa, puoi creare un CloudTrail Lake, che conserva i log di CloudTrail per un massimo di sette anni e fornisce una funzionalità di query basata su SQL.

  AWS consiglia ai clienti che utilizzano VPC di attivare il traffico di rete e i log DNS utilizzando rispettivamente log di flusso VPC e log delle query del risolutore Amazon Route 53 e di inviarli in streaming su un bucket Amazon S3 o un gruppo di log CloudWatch. Il log di flusso VPC può essere creato per un VPC, una sottorete o un'interfaccia di rete. Per i log di flusso VPC, puoi scegliere come e dove utilizzarli per ridurre i costi.

  I log AWS CloudTrail, i log di flusso VPC e i log delle query del risolutore Route 53 sono le origini dei log di base per supportare le indagini sulla sicurezza in AWS. Puoi anche utilizzare Amazon Security Lake per raccogliere, normalizzare e archiviare questi dati di log in formato Apache Parquet e Open Cybersecurity Schema Framework (OCSF), pronto per la query. Security Lake supporta anche altri log AWS e log provenienti da origini di terze parti.

  I servizi AWS possono generare log non acquisiti dalle origini di log di base, come log di Elastic Load Balancing, log di AWS WAF, log del registratore AWS Config, esiti di Amazon GuardDuty, log di audit di Amazon Elastic Kubernetes Service (Amazon EKS) e log del sistema operativo e delle applicazioni delle istanze Amazon EC2. Per un elenco completo delle opzioni di log e monitoraggio, consulta l'Appendice A: definizioni delle capacità del cloud, log ed eventi della AWS Security Incident Response Guide.
- Funzionalità di log delle ricerche per ogni servizio e applicazioneAWS: ciascun servizio e applicazione AWS offre opzioni per l'archiviazione di log, ciascuna con le proprie funzionalità relative a conservazione e ciclo di vita. I due servizi di archiviazione di log più comuni sono Amazon

Simple Storage Service (Amazon S3) e Amazon CloudWatch. Per lunghi periodi di conservazione, è consigliabile utilizzare Amazon S3 per la sua convenienza in termini di costi e per la flessibilità del ciclo di vita. Se l'opzione principale di log è Amazon CloudWatch Logs, puoi prendere in considerazione l'archiviazione dei log ad accesso meno frequente su Amazon S3.

- Seleziona l'archiviazione dei log: la scelta dell'archiviazione dei log è in genere correlata allo strumento di query utilizzato, alle funzionalità di conservazione, alla conoscenza e ai costi. Le opzioni principali per il log storage sono un bucket Amazon S3 o un gruppo di log CloudWatch.

  Un bucket Amazon S3 offre a possibilità di un'archiviazione economica e duratura, con una policy opzionale per il ciclo di vita. È possibile eseguire query sui log archiviati nei bucket Amazon S3 mediante servizi come Amazon Athena.

  Un gruppo di log di CloudWatch offre un'archiviazione durevole e una funzione di query integrata attraverso Approfondimenti di CloudWatch Logs.

- Identifica la conservazione dei log adeguata: se utilizzi un bucket Amazon S3 o un gruppo di log CloudWatch per archiviare i log, stabilisci cicli di vita adeguati per ogni origine di log al fine di ottimizzare i costi di archiviazione e recupero. In genere i clienti hanno a disposizione da tre mesi a un anno di log per le query, con una conservazione fino a sette anni. La scelta di disponibilità e conservazione deve essere in linea con i requisiti di sicurezza e con un insieme di mandati statutari, normativi e aziendali.
- Utilizza la registrazione per ciascun servizio e applicazione AWS con policy di conservazione e ciclo di vita adeguate: per ciascun servizio o applicazione AWS della tua organizzazione, consulta le linee guida specifiche sulla configurazione dei log:
  - Configurazione di AWS CloudTrail Trail
  - Configurazione di log di flusso VPC
  - Configurazione dell'esportazione degli esiti di Amazon GuardDuty
  - Configurazione delle registrazioni AWS Config
  - Configurazione del traffico ACL web di AWS WAF
  - Configurazione dei log del traffico di rete di AWS Network Firewall
  - Configurazione dei log di accesso per Elastic Load Balancing
  - Configurazione del log delle query del risolutore Amazon Route 53
  - Configurazione dei log di Amazon RDS
  - Configurazione dei log del piano di controllo (control-plane) di Amazon EKS
  - Configurazione dell'agente Amazon CloudWatch per istanze Amazon EC2 e server on-premises

- Seleziona e implementa meccanismi di query per i log: per le query dei log, puoi utilizzare Approfondimenti di CloudWatch Logs per i dati archiviati nei gruppi di log di CloudWatch, Amazon Athena e il Servizio OpenSearch di Amazon per i dati archiviati in Amazon S3. Inoltre, puoi utilizzare strumenti di query di terze parti, come un servizio di gestione delle informazioni e degli eventi di sicurezza (SIEM).

  Il processo di selezione di uno strumento di query dei log deve considerare gli aspetti relativi a persone, processi e tecnologia delle operazioni di sicurezza. Occorre scegliere uno strumento che soddisfi i requisiti operativi, aziendali e di sicurezza, accessibile e di cui sia possibile effettuare la manutenzione a lungo termine. Tieni presente che gli strumenti di query dei log funzionano in modo ottimale quando il numero di log da analizzare è mantenuto entro i limiti dello strumento. Non è raro avere più strumenti di query a causa di vincoli tecnici o di costo.

  Ad esempio, puoi ricorrere a uno strumento di gestione delle informazioni e degli eventi di sicurezza (SIEM) di terze parti per eseguire query sugli ultimi 90 giorni di dati, ma utilizzare Athena per eseguire query oltre i 90 giorni a causa dei costi di importazione dei log di un SIEM. Indipendentemente dall'implementazione, verifica che il tuo approccio riduca al minimo il numero di strumenti necessari per ottimizzare l'efficienza operativa, soprattutto durante le indagini su un evento di sicurezza.
- Usa i log per gli avvisi: AWS fornisce avvisi tramite diversi servizi di sicurezza:
  - AWS Config monitora e registra le configurazioni delle risorse AWS e consente di automatizzare la valutazione e la correzione rispetto alle configurazioni desiderate.
  - Amazon GuardDuty è un servizio di rilevamento delle minacce che esegue un monitoraggio continuo per individuare attività dannose e comportamenti non autorizzati al fine di proteggere carichi di lavoro e Account AWS. GuardDuty acquisisce, aggrega e analizza le informazioni dalle origini, come eventi di gestione e dati AWS CloudTrail, log DNS, log di flusso VPC e log di audit di Amazon EKS. GuardDuty estrae flussi di dati indipendenti direttamente da CloudTrail, log di flussi VPC, log di query DNS e Amazon EKS. Non è necessario gestire le policy del bucket Amazon S3 o modificare le modalità di raccolta e archiviazione dei log. È comunque consigliabile mantenere questi log a fini investigativi e di conformità.
  - AWS Security Hub CSPM offre un unico punto di aggregazione, organizzazione e assegnazione di priorità per gli avvisi di sicurezza o gli esiti provenienti da diversi servizi AWS e da prodotti opzionali di terze parti per fornire una panoramica completa degli avvisi di sicurezza e dello stato di conformità.

  Esistono anche motori di generazione di avvisi personalizzati per gli avvisi di sicurezza non coperti da questi servizi o per gli avvisi specifici relativi al tuo ambiente. Per informazioni sulla creazione

di questi avvisi e rilevamenti, consulta la sezione Detection nella AWS Security Incident Response Guide.

Risorse

Best practice correlate:

- SEC04-BP02 Acquisizione di log, esiti e metriche in posizioni standardizzate
- SEC07-BP04 Definizione della gestione del ciclo di vita dei dati scalabili
- SEC10-BP06 Implementazione anticipata degli strumenti

Documenti correlati:

- AWS Security Incident Response Guide
- Nozioni di base su Amazon Security Lake
- Nozioni di base su Amazon CloudWatch Logs

Video correlati:

- AWS re:Invent 2022 - Introducing Amazon Security Lake

Esempi correlati:

- Assisted Log Enabler for AWS
- AWS Security Hub CSPM Findings Historical Export

SEC04-BP02 Acquisizione di log, esiti e metriche in posizioni standardizzate

I team di sicurezza si basano su log ed esiti per analizzare gli eventi che possono indicare attività non autorizzate o modifiche non intenzionali. Per semplificare tale analisi, acquisisci i log e gli esiti di sicurezza in posizioni standardizzate. Ciò rende disponibili i punti di interesse dei dati per la correlazione e può semplificare le integrazioni degli strumenti.

Risultato desiderato: un approccio standardizzato alla raccolta, analisi e visualizzazione di dati di log, esiti e metriche. I team di sicurezza possono correlare, analizzare e visualizzare in modo efficiente i dati di sicurezza su sistemi diversi per scoprire potenziali eventi di sicurezza e identificare le

anomalie. I sistemi di gestione delle informazioni e degli eventi di sicurezza (SIEM) o altri meccanismi sono integrati per effettuare query e analizzare i dati dei log per risposte tempestive, tracciare ed eseguire escalation degli eventi di sicurezza.

Anti-pattern comuni:

- I team hanno e gestiscono in modo indipendente la raccolta di log e metriche che non è coerente con la strategia di registrazione dell'organizzazione.
- I team non dispongono di controlli di accesso adeguati per limitare visibilità e alterazione dei dati raccolti.
- I team non gestiscono log, esiti e metriche di sicurezza nell'ambito della loro policy di classificazione dei dati.
- I team trascurano i requisiti di sovranità e localizzazione dei dati durante la configurazione delle raccolte di dati.

Vantaggi dell'adozione di questa best practice: una soluzione di log standardizzata per raccogliere ed effettuare query su dati ed eventi dei log garantisce approfondimenti migliori ricavati dalle informazioni in essi contenute La configurazione di un ciclo di vita automatizzato per i dati di log raccolti può ridurre i costi sostenuti per l'archiviazione dei log. È possibile creare un controllo degli accessi granulare per le informazioni di log raccolte, in base a sensibilità dei dati e modelli di accesso richiesti dai team. Puoi integrare strumenti per correlare, visualizzare e ricavare informazioni dai dati.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

La crescita dell'utilizzo di AWS all'interno di un'organizzazione comporta un numero crescente di carichi di lavoro e ambienti distribuiti. Dato che ciascuno di questi carichi di lavoro e ambienti genera dati sull'attività al suo interno, l'acquisizione e l'archiviazione di questi dati a livello locale rappresenta una sfida per le operazioni di sicurezza. I team addetti alla sicurezza utilizzano strumenti come i sistemi di gestione delle informazioni e degli eventi di sicurezza (SIEM) per raccogliere dati da origini distribuite e sottoporli a flussi di lavoro di correlazione, analisi e risposta. Ciò richiede la gestione di una serie complessa di autorizzazioni per l'accesso alle varie origini dati e un sovraccarico aggiuntivo nel funzionamento dei processi di estrazione, trasformazione e caricamento (ETL).

Per superare queste sfide, valuta la possibilità di aggregare tutte le origini pertinenti dei dati dei log di sicurezza in un account Log Archive, come illustrato in Organizing Your AWS Environment Using Multiple Accounts. Ciò comprende tutti i dati relativi alla sicurezza provenienti da carichi di

lavoro e log generati dai servizi AWS, come AWS CloudTrail, AWS WAF, Elastic Load Balancing e Amazon Route 53. L'acquisizione di questi dati in posizioni standardizzate e in un Account AWS separato con autorizzazioni tra account adeguate presenta diversi vantaggi. Questa pratica aiuta a prevenire la manomissione dei log all'interno di ambienti e carichi di lavoro compromessi, fornisce un unico punto di integrazione per strumenti aggiuntivi, oltre a offrire un modello più semplificato per la configurazione della conservazione dei dati e del ciclo di vita. Valuta gli impatti della sovranità dei dati, degli ambiti di conformità e di altre normative per determinare se sono necessarie più sedi di archiviazione di dati di sicurezza e relativi periodi di conservazione.

Per semplificare acquisizione e standardizzazione di log ed esiti, prendi in considerazione Amazon Security Lake nel tuo account Log Archive. Puoi configurare Security Lake in modo che importi in automatico dati da origini comuni come CloudTrail, Route 53, Amazon EKS e flussi di log VPC. Puoi anche configurare AWS Security Hub CSPM come origine dati in Security Lake, in modo da correlare gli esiti di altri servizi AWS, come Amazon GuardDuty e Amazon Inspector, con i tuoi dati di log. Puoi anche utilizzare integrazioni di origini dati di terze parti o configurare origini dati personalizzate. Tutte le integrazioni standardizzano i dati nel formato Open Cybersecurity Schema Framework (OCSF) e la relativa archiviazione avviene in bucket Amazon S3 come file Parquet, così da eliminare la necessità di elaborazione ETL.

L'archiviazione dei dati di sicurezza in posizioni standardizzate offre funzionalità di analisi avanzate. AWS consiglia di implementare strumenti per l'analisi della sicurezza operanti in un ambiente AWS in un account Security Tooling separato dal proprio account Log Archive. Questo approccio consente di implementare controlli approfonditi per proteggere l'integrità e la disponibilità dei log e del processo di gestione dei log, distinti dagli strumenti che vi accedono. Prendi in considerazione l'utilizzo di servizi, come Amazon Athena, per l'esecuzione di query su richiesta che mettono in correlazione più origini dati. Puoi anche integrare strumenti di visualizzazione, come Quick. Le soluzioni basate sull'intelligenza artificiale sono sempre più disponibili e possono svolgere funzioni quali la traduzione degli esiti in sintesi leggibili dall'uomo e l'interazione in linguaggio naturale. Queste soluzioni sono spesso più facilmente integrate grazie a una posizione di archiviazione di dati standardizzata per le interrogazioni.

Passaggi dell'implementazione

1. Crea gli account di archiviazione di log e Security Tooling
    a. Mediante AWS Organizations, crea gli account Log Archive e Security Tooling in un'unità organizzativa di sicurezza. Se utilizzi AWS Control Tower per gestire la tua organizzazione, gli account Log Archive e Security Tooling vengono creati in automatico. Configura ruoli e autorizzazioni per l'accesso a questi account e la loro amministrazione, come richiesto.

2. Configurazione delle posizioni standardizzate dei dati di sicurezza
    a. Determina la tua strategia per la creazione di posizioni di dati di sicurezza standardizzate. Puoi raggiungere questo obiettivo mediante opzioni come approcci architetturali comuni per i data lake, prodotti per dati di terze parti o Amazon Security Lake. AWS consiglia di acquisire i dati di sicurezza da Regioni AWS che hai specificato per i tuoi account, anche in caso di mancato utilizzo attivo.
3. Configura la pubblicazione delle origini dati nelle tue posizioni standardizzate
    a. Identifica le origini per i tuoi dati di sicurezza e configurale per la pubblicazione in posizioni standardizzate. Valuta le opzioni per l'esportazione automatica dei dati nel formato desiderato anziché in quelle in cui è necessario sviluppare processi ETL. Amazon Security Lake ti consente di raccogliere dati da origini AWS supportate e sistemi integrati di terze parti.
4. Configura gli strumenti per l'accesso alle tue posizioni standardizzate
    a. Configura strumenti come Amazon Athena, QuickSight o soluzioni di terze parti per disporre dell'accesso necessario alle tue posizioni standardizzate. Configura questi strumenti in modo che operino dall'account Security Tooling con accesso in lettura trasversale all'account Log Archive, se applicabile. Crea abbonati in Amazon Security Lake così da fornire a questi strumenti l'accesso ai dati.

Risorse

Best practice correlate:

- SEC01-BP01 Separazione dei carichi di lavoro tramite account
- SEC07-BP04 Definizione della gestione del ciclo di vita dei dati
- SEC08-BP04 Applicazione del controllo degli accessi
- OPS08-BP02 Analizza i log relativi ai carichi di lavoro

Documenti correlati:

- Whtepaper AWS: Organizing Your AWS Environment Using Multiple Accounts
- Guida prescrittiva AWS: AWS Security Reference Architecture (AWS SRA)
- AWS Guida prescrittiva : Logging and monitoring guide for application owners

Esempi correlati:

- Aggregazione, ricerca e visualizzazione dei dati di log da origini distribuite con Amazon Athena e QuickSight
- Come visualizzare gli esiti di Amazon Security Lake con QuickSight
- Generate AI powered insights for Amazon Security Lake using Amazon SageMaker AI Studio and Amazon Bedrock
- Identify cybersecurity anomalies in your Amazon Security Lake data using Amazon SageMaker
- Ingest, transform, and deliver events published by Amazon Security Lake to Amazon OpenSearch Service
- Simplify AWS CloudTrail log analysis with natural language query generation in CloudTrail Lake

Strumenti correlati:

- Amazon Security Lake
- Integrazioni con i partner di Amazon Security Lake
- Open Cybersecurity Schema Framework (OCSF)
- Amazon Athena
- Rapidità
- Amazon Bedrock

SEC04-BP03 Correlazione e arricchimento degli avvisi di sicurezza

Un'attività imprevista può generare diversi avvisi di sicurezza da origini diverse, richiedendo un'ulteriore correlazione e arricchimento per la comprensione del contesto completo. Implementa correlazione e arricchimento automatizzati degli avvisi di sicurezza per un'identificazione e una risposta agli incidenti più accurate.

Risultato desiderato: mentre l'attività generano avvisi diversi all'interno di carichi di lavoro e ambienti, i meccanismi automatizzati correlano i dati e li arricchiscono con informazioni aggiuntive. Questa pre-elaborazione presenta un quadro più dettagliato dell'evento, che aiuta gli investigatori a determinare la criticità dell'evento e a stabilire se si tratta di un incidente che richiede una risposta formale. Questo processo riduce il carico sui team di monitoraggio e investigazione.

Anti-pattern comuni:

- Gruppi diversi di persone esaminano esiti e avvisi generati da sistemi differenti, a meno che i requisiti di separazione degli incarichi non impongano altrimenti.

- L'organizzazione convoglia tutti i dati di esiti e avvisi di sicurezza in posizioni standard, ma richiede agli investigatori di eseguire correlazioni e arricchimenti manuali.
- Ti affidi esclusivamente all'intelligence dei sistemi di rilevamento delle minacce per riferire sugli esiti e stabilire la criticità.

Vantaggi dell'adozione di questa best practice: riduzione del carico cognitivo complessivo e della preparazione manuale dei dati richiesta agli investigatori grazie a correlazione e arricchimento automatizzati degli avvisi. Questa pratica può ridurre il tempo necessario per determinare se l'evento rappresenta un incidente e avviare una risposta formale. Un contesto aggiuntivo consente inoltre di valutare con precisione la reale gravità di un evento, in quanto può essere superiore o inferiore a quanto suggerito da un avviso.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Gli avvisi di sicurezza possono provenire da diverse sorgenti all'interno di AWS, tra cui:

- Servizi come Amazon GuardDuty, AWS Security Hub CSPM, Amazon Macie, Amazon Inspector, AWS Config, AWS Identity and Access Management Access Analyzer e Strumento di analisi degli accessi alla rete
- Avvisi provenienti dall'analisi automatizzata dei log di servizi, infrastrutture e applicazioni AWS, ad esempio da Security Analytics per il Servizio OpenSearch di Amazon.
- Allarmi in risposta a modifiche nella tua attività di fatturazione provenienti da origini come Amazon CloudWatch, Amazon EventBridge o Budget AWS.
- Origini di terze parti come feed di intelligence sulle minacce e Soluzioni dei partner per la sicurezza da AWS Partner Network
- Contatto tramite AWS Trust & Safety o altre origini, come clienti o dipendenti interni.
- Utilizza Threat Technique Catalog by AWS (TTC) per facilitare l'identificazione e la correlazione del comportamento degli autori delle minacce attraverso l'identificazione degli indicatori di compromissione (IoC). Il TTC è un'estensione del framework MITRE ATT&CK, che classifica tutti i comportamenti e le tecniche noti e osservati degli autori di minacce rivolti alle risorse AWS.

Nella loro forma più elementare, gli avvisi contengono informazioni su chi (il principale o l'identità) sta facendo cosa (l'azione intrapresa) e cosa (le risorse interessate). Per ognuna di queste origini, individua le modalità con cui puoi creare mappature tra gli identificatori per queste identità, azioni e

risorse come base per eseguire la correlazione. Ciò può avvenire integrando le origini degli avvisi con uno strumento di gestione delle informazioni e degli eventi di sicurezza (SIEM) per eseguire la correlazione automatica, creando pipeline ed elaborazioni di dati proprie o una combinazione di entrambi.

Un esempio di servizio in grado di eseguire la correlazione è Amazon Detective. Il rilevatore inserisce continuamente avvisi da varie origini AWS e da terze parti e utilizza diverse forme di intelligenza per creare un grafico visivo delle loro relazioni in modo da semplificare le indagini.

Sebbene la criticità iniziale di un avviso sia un aiuto per la definizione delle priorità, il relativo contesto di generazione ne determina la vera criticità. Ad esempio, Amazon GuardDuty può mostrare avvisi che indicano che un'istanza Amazon EC2 all'interno del tuo carico di lavoro sta eseguendo una query su un nome di dominio inaspettato. GuardDuty potrebbe assegnare una bassa criticità a questo avviso. Tuttavia, la correlazione automatica con altre attività svolte al momento dell'allarme potrebbe rivelare che diverse centinaia di istanze EC2 sono state distribuite dalla stessa identità, con un conseguente aumento dei costi operativi complessivi. In tal caso, questo contesto di eventi correlati causerebbe la visualizzazione di un nuovo avviso di sicurezza, la cui criticità potrebbe essere regolata su un livello superiore accelerando così l'esecuzione di ulteriori azioni.

Passaggi dell'implementazione

1. Identifica le origini delle informazioni sugli avvisi di sicurezza. Scopri come gli avvisi provenienti da questi sistemi rappresentano identità, azioni e risorse per determinare dove è possibile una correlazione.
2. Stabilisci un meccanismo per acquisire avvisi da diverse origini. Prendi in considerazione servizi come Security Hub CSPM, EventBridge e CloudWatch a tale scopo.
3. Identifica le origini per correlazione e arricchimento dei dati. Alcuni esempi di origini sono: AWS CloudTrail, log di flusso VPC, log del risolutore Route 53 e log di infrastrutture e applicazioni. Alcuni di questi log, oppure tutti, potrebbero essere utilizzati tramite un'unica integrazione con Amazon Security Lake.
4. Integra i tuoi avvisi con le tue origini di correlazione e arricchimento dei dati per creare contesti degli eventi di sicurezza più dettagliati e stabilire le criticità.
    a. Amazon Detective, strumenti SIEM o altre soluzioni di terze parti possono eseguire in automatico un determinato livello di inserimento, correlazione e arricchimento.
    b. Puoi anche utilizzare i servizi AWS per crearne uno tuo. Ad esempio, puoi richiamare una funzione AWS Lambda per eseguire una query Amazon Athena rispetto a AWS CloudTrail o Amazon Security Lake e pubblicare i risultati su EventBridge.

Risorse

Best practice correlate:

- SEC10-BP03 Preparazione di funzionalità forensi
- OPS08-BP04 Creare avvisi fruibili
- REL06-BP03 Invio di notifiche (elaborazione e avvisi in tempo reale)

Documenti correlati:

- AWS Security Incident Response Guide

Esempi correlati:

- How to enrich AWS Security Hub CSPM findings with account metadata

Strumenti correlati:

- Amazon Detective
- Amazon EventBridge
- AWS Lambda
- Amazon Athena

SEC04-BP04 Avvio della riparazione delle risorse non conformi

I controlli investigativi possono segnalare la presenza di risorse non conformi ai requisiti di configurazione. È possibile avviare interventi correttivi definiti in modo programmatico, sia manualmente sia automaticamente, per riparare queste risorse e ridurre al minimo gli impatti potenziali. Quando definisci le correzioni in modo programmatico, puoi intraprendere azioni rapide e coerenti.

Sebbene l'automazione possa migliorare le operazioni di sicurezza, occorre implementarla e gestirla con attenzione. Implementa meccanismi di supervisione e controllo opportuni per verificare che le risposte automatizzate siano efficaci, accurate e in linea con le policy organizzative e la propensione al rischio.

Risultato desiderato: definizione di standard di configurazione delle risorse insieme a passaggi correttivi in caso di rilevamento di una mancata conformità. Dove possibile, hai definito gli interventi correttivi in modo programmatico, in modo da avviarli manualmente o attraverso l'automazione. Sono disponibili sistemi di rilevamento per identificare le risorse non conformi e pubblicare avvisi in strumenti centralizzati monitorati dal personale di sicurezza. Questi strumenti supportano l'esecuzione degli interventi correttivi programmatici, manualmente o automaticamente. Le soluzioni automatiche dispongono di meccanismi di supervisione e controllo adeguati per regolarne l'utilizzo.

Anti-pattern comuni:

- Automazione implementata, ma non si riescono a testare e convalidare a fondo le azioni correttive. Ciò può comportare conseguenze indesiderate, come l'interruzione delle operazioni aziendali legittime o l'instabilità del sistema.
- L'automazione migliora tempi e procedure di risposta, ma senza un monitoraggio adeguato e senza meccanismi che consentano l'intervento umano e la valutazione, quando necessario.
- Ci si affida esclusivamente agli interventi correttivi, senza considerarli come parte di un programma più ampio di risposta agli incidenti e di ripristino.

Vantaggi dell'adozione di questa best practice: gli interventi correttivi automatici possono rispondere alle configurazioni errate più rapidamente rispetto ai processi manuali, il che contribuisce a ridurre al minimo i potenziali impatti aziendali e a ridurre la finestra di opportunità per usi indesiderati. Nel definire gli interventi correttivi in modo programmatico, questi vengono applicate in modo coerente, il che riduce il rischio di errore umano. L'automazione è altresì in grado di gestire un volume maggiore di avvisi contemporaneamente, il che è molto importante negli ambienti che operano su larga scala.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Come illustrato in SEC01-BP03 Identificazione e convalida degli obiettivi di controllo, servizi come AWS Config ed AWS Security Hub CSPM aiutano a monitorare la configurazione delle risorse nei tuoi account per verificarne la conformità ai tuoi requisiti. Quando vengono rilevate risorse non conformi, servizi come AWS Security Hub CSPM possono aiutare a instradare gli avvisi in modo appropriato e ad apportare le correzioni necessarie. Queste soluzioni offrono agli investigatori della sicurezza il punto centrale per il monitoraggio dei problemi e l'adozione di misure correttive.

Oltre a AWS Security Hub CSPM, AWS ha introdotto Security Hub Advanced. Questo servizio, annunciato al re:Invent 2025, trasforma il modo in cui le organizzazioni danno priorità ai problemi di

sicurezza più critici e rispondono su larga scala per proteggere i propri ambienti cloud. Il Security Hub avanzato ora utilizza analisi avanzate per correlare, arricchire e dare priorità in automatico ai segnali di sicurezza in tutto l'ambiente cloud. Security Hub si integra perfettamente con Amazon GuardDuty, Amazon Inspector, Amazon Macie e AWS Security Hub CSPM Gli esiti correlati in Security Hub possono portare a un nuovo esito, chiamato esito di esposizione, che include un presunto percorso di attacco basato sulle vulnerabilità rilevate in ciascuna risorsa.

Mentre alcune situazioni di non conformità delle risorse sono uniche e la loro risoluzione richiede il giudizio umano, altre situazioni hanno una risposta standard che si può definire in maniera programmatica. Ad esempio, una risposta standard a un gruppo di sicurezza VPC configurato in modo errato potrebbe consistere nella rimozione delle regole non consentite e della notifica al proprietario. È possibile definire le risposte nelle funzioni di AWS Lambda, nei documenti di AWS Systems Manager Automation o tramite altri ambienti di codice di propria preferenza. Assicurati che l'ambiente sia in grado di autenticarsi ad AWS utilizzando un ruolo IAM con il minor numero di autorizzazioni necessarie per intraprendere un'azione correttiva.

Una volta definita la correzione desiderata, è possibile determinare i mezzi preferiti per avviarla. AWS Config può avviare le azioni correttive per tuo conto. Se utilizzi Security Hub CSPM, puoi farlo tramite le azioni personalizzate, che pubblicano le informazioni sugli esiti in Amazon EventBridge. Una regola EventBridge può quindi avviare l'azione correttiva. Puoi configurare le correzioni tramite Security Hub CSPM in modo che l'esecuzione sia automatica o manuale.

Per le azioni correttive programmatiche, ti consigliamo di disporre di log e audit completi delle azioni intraprese e dei relativi risultati. Rivedi e analizza questi log per valutare l'efficacia dei processi automatizzati e identificare le aree di miglioramento. Acquisisci i log in Amazon CloudWatch Logs e i risultati delle azioni correttive sotto forma di note sugli esiti in Security Hub CSPM.

Parti prendendo in considerazione la risposta di sicurezza automatizzata su AWS, che offre soluzioni predefinite per risolvere gli errori di configurazione di sicurezza più comuni.

Passaggi dell'implementazione

1. Analizza e assegna priorità agli avvisi.
    a. Consolida gli avvisi di sicurezza provenienti da vari servizi AWS in Security Hub CSPM per una visibilità, una definizione delle priorità e una correzione centralizzate.
2. Sviluppa soluzioni correttive.
    a. Utilizza servizi come Systems Manager e AWS Lambda per eseguire correzioni programmatiche.

3. Configura le modalità di avvio delle correzioni.
    a. Utilizzando Systems Manager, definisci le azioni personalizzate che pubblicano gli esiti su EventBridge. Configura queste azioni in modo l'avvio avvenga manualmente o automaticamente.
    b. Puoi anche utilizzare Amazon Simple Notification Service (SNS) per inviare notifiche e avvisi alle parti interessate (come il team di sicurezza o i team di risposta agli incidenti) per l'intervento manuale o l'escalation, laddove necessario.
4. Rivedi e analizza i log delle correzioni per verificarne efficacia e miglioramenti.
    a. Invia l'output del log a CloudWatch Logs. Acquisisci i risultati come note sull'esito in Security Hub CSPM.

Risorse

Best practice correlate:

- SEC06-BP03 Riduzione della gestione manuale e dell'accesso interattivo

Documenti correlati:

- AWS Security Incident Response Guide - Detection

Esempi correlati:

- Risposta di sicurezza automatizzata su AWS
- Monitor EC2 instance key pairs using AWS Config
- Create AWS Config custom rules by using AWS CloudFormation Guard policies
- Automatically remediate unencrypted Amazon RDS DB instances and clusters

Strumenti correlati:

- AWS Systems Manager Automation
- Risposta di sicurezza automatizzata su AWS

Protezione dell’infrastruttura

Questions

- SEC 5. In che modo proteggi le risorse di rete?
- SEC 6. In che modo proteggi le risorse di calcolo?

SEC 5. In che modo proteggi le risorse di rete?

Qualsiasi carico di lavoro che abbia una qualche forma di connettività di rete, che si tratti di Internet o di una rete privata, richiede più livelli di difesa per proteggere da minacce esterne e interne basate sulla rete.

Best practice

- SEC05-BP01 Creazione di livelli di rete
- SEC05-BP02 Controllo del traffico a tutti i livelli
- SEC05-BP03 Implementare la protezione basata sull'ispezione
- SEC05-BP04 Automatizza la protezione della rete

SEC05-BP01 Creazione di livelli di rete

Segmenta la topologia di rete in diversi livelli basati su raggruppamenti logici dei componenti del carico di lavoro in base alla sensibilità dei dati e ai requisiti di accesso. Distingui tra i componenti che richiedono l’accesso in entrata da Internet, come gli endpoint Web pubblici, e quelli che necessitano solo di un accesso interno, come i database.

Risultato desiderato: i livelli della rete rientrano in un approccio di difesa approfondito e integrale alla sicurezza che integra l’autenticazione delle identità e la strategia di autorizzazione dei carichi di lavoro. I livelli sono implementati in base alla sensibilità dei dati e ai requisiti di accesso, con meccanismi adeguati in termini di flusso e controllo del traffico.

Anti-pattern comuni:

- Creazione di tutte le risorse in un VPC o una sottorete unica.
- Creazione dei livelli di rete senza considerare i requisiti di sensibilità dei dati, il comportamento dei componenti o la loro funzionalità.
- Utilizzo di VPC e sottoreti come impostazioni predefinite per tutte le considerazioni relative al livello di rete senza considerare come i servizi gestiti da AWS influenzino la tua topologia.

Vantaggi dell'adozione di questa best practice: la definizione di livelli di rete è il primo passo per limitare i percorsi superflui lungo la rete, in particolare quelli che conducono a sistemi e dati critici. In tal modo gli attori non autorizzati avranno più difficoltà ad accedere alla rete e a navigare verso altre risorse al suo interno. I livelli di rete discreti riducono l'ambito di analisi dei sistemi di ispezione, ad esempio per il rilevamento delle intrusioni o la prevenzione del malware. Di conseguenza, si riduce il potenziale di falsi positivi e il sovraccarico di elaborazione non necessario.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Quando si progetta l'architettura di un carico di lavoro, è comune separare i componenti in diversi livelli in base alle rispettive responsabilità. Ad esempio, un'applicazione Web può avere un livello di presentazione, uno di applicazione e uno di dati. È possibile adottare un approccio simile quando progetti la tua topologia di rete. I controlli di rete sottostanti possono contribuire a far rispettare i requisiti di accesso ai dati del carico di lavoro. Ad esempio, in un'architettura di applicazioni Web a tre livelli, puoi archiviare i file statici del livello di presentazione su Amazon S3 e distribuirli da una rete di distribuzione di contenuti (CDN), come Amazon CloudFront. Il livello di applicazione può avere endpoint pubblici che un Application Load Balancer (ALB) distribuisce in una sottorete pubblica Amazon VPC (simile a una zona demilitarizzata o DMZ), con servizi di backend implementati in sottoreti private. Il livello dati che funge da host per risorse come database e file system condivisi può risiedere in sottoreti private diverse dalle risorse del livello applicativo. In corrispondenza di ciascuno di questi limiti di livello (CDN, sottorete pubblica, sottorete privata), è possibile implementare controlli che consentano solo al traffico autorizzato di attraversarli.

Analogamente alla modellazione dei livelli di rete in base allo scopo funzionale dei componenti del carico di lavoro, occorre prendere in considerazione anche la sensibilità dei dati elaborati. Utilizzando l'esempio dell'applicazione Web, mentre tutti i servizi del carico di lavoro possono risiedere all'interno del livello di applicazione, servizi diversi possono elaborare dati con livelli di sensibilità differenti. In questo caso, la divisione del livello di applicazione utilizzando più sottoreti private, diversi VPC nello stesso Account AWS o persino VPC diversi in diversi Account AWS per ciascun livello di sensibilità dei dati può essere adeguata in base ai requisiti di controllo.

Un'ulteriore considerazione per i livelli di rete consiste nella coerenza del comportamento dei componenti del carico di lavoro. Continuando con l'esempio, nel livello di applicazione possono essere presenti servizi che accettano input dagli utenti finali o integrazioni di sistemi esterni intrinsecamente più rischiosi rispetto agli input di altri servizi. A titolo esemplificativo, si possono citare il caricamento di file, l'esecuzione di script di codice, la scansione di e-mail e così via. La collocazione di questi servizi nel proprio livello di rete contribuisce a creare un limite di isolamento più forte attorno

a essi e può evitare che il loro comportamento unico crei falsi positivi in termini di allarmi nei sistemi di ispezione.

Nell'ambito della progettazione, prendi in considerazione in che modo l'utilizzo dei servizi AWS gestiti influenza la topologia di rete. Scopri in che modo servizi come Amazon VPC Lattice semplificano l'interoperabilità dei componenti del carico di lavoro tra i livelli di rete. In caso di utilizzo di AWS Lambda, esegui l'implementazione nelle sottoreti VPC, salvo in presenza di motivazioni contrarie specifiche. Determina dove si trovano gli endpoint VPC e semplifica con AWS PrivateLink il rispetto delle policy di sicurezza che limitano l'accesso ai gateway Internet.

Passaggi dell'implementazione

1. Rivedi l'architettura del carico di lavoro. Raggruppa in modo logico componenti e servizi in base alle funzioni che svolgono, alla sensibilità dei dati elaborati e al loro comportamento.
2. Per i componenti che rispondono alle richieste provenienti da Internet, prendi in considerazione l'utilizzo di bilanciatori del carico o altri proxy per fornire endpoint pubblici. Esamina il trasferimento dei controlli di sicurezza utilizzando servizi gestiti, come CloudFront, Gateway Amazon API, Elastic Load Balancing e AWS Amplify per l'hosting di endpoint pubblici.
3. I componenti in esecuzione in ambienti di calcolo, come istanze Amazon EC2, container AWS Fargate o funzioni Lambda, vanno implementati in sottoreti private, basate sui tuoi gruppi sin dal primo passaggio.
4. Per servizi AWS completamente gestiti, come Amazon DynamoDB, Amazon Kinesis o Amazon SQS, prendi in considerazione l'utilizzo degli endpoint VPC come impostazione predefinita per l'accesso tramite indirizzi IP privati.

Risorse

Best practice correlate:

- REL02 Come si pianifica la topologia di rete?
- PERF04-BP01 In che modo la rete influisce sulle prestazioni

Video correlati:

- AWS re:Invent 2023 - AWS networking foundations

Esempi correlati:

- Esempi di VPC
- Access container applications privately on Amazon ECS by using AWS Fargate, AWS PrivateLink, and a Network Load Balancer
- Serve static content in an Amazon S3 bucket through a VPC by using Amazon CloudFront

SEC05-BP02 Controllo del traffico a tutti i livelli

All'interno dei livelli della rete, utilizza un'ulteriore segmentazione per limitare il traffico solo ai flussi necessari per ogni carico di lavoro. In primo luogo, concentrati sul controllo del traffico tra Internet o altri sistemi esterni verso un carico di lavoro e il tuo ambiente (traffico nord-sud). Quindi, esamina i flussi tra diversi componenti e sistemi (traffico est-ovest).

Risultato desiderato: solo i flussi di rete necessari ai componenti dei tuoi carichi di lavoro possono comunicare tra loro e con i rispettivi client e con qualsiasi altro servizio da cui dipendono. La tua progettazione tiene conto di considerazioni come l'ingresso e l'uscita pubblici rispetto a quelli privati, la classificazione dei dati, le normative regionali e i requisiti di protocollo. Laddove possibile, preferisci flussi punto a punto rispetto al peering di rete come parte della progettazione secondo il principio del privilegio minimo.

Anti-pattern comuni:

- Adozione di un approccio alla sicurezza della rete basato sul perimetro e controllare il flusso di traffico solo al confine dei livelli di rete.
- Si presume che tutto il traffico all'interno di un livello di rete sia autenticato e autorizzato.
- Applicazione dei controlli al traffico in ingresso o a quello in uscita, ma non a entrambi.
- Affidamento esclusivo per l'autenticazione e l'autorizzazione del traffico ai componenti del carico di lavoro e ai controlli di rete.

Vantaggi dell'adozione di questa best practice: questa pratica consente di ridurre il rischio di movimenti non autorizzati all'interno della rete e aggiunge un ulteriore livello di autorizzazione ai carichi di lavoro. Eseguendo il controllo del flusso di traffico, è possibile limitare la portata dell'impatto di un incidente di sicurezza e velocizzare il rilevamento e la risposta.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Se da un lato i livelli di rete aiutano a stabilire i limiti dei componenti del carico di lavoro che presentano una funzione, un livello di sensibilità dei dati e un comportamento simili, dall'altro è possibile creare un livello di controllo del traffico molto più granulare utilizzando tecniche per segmentare ulteriormente i componenti all'interno di questi livelli, seguendo il principio del privilegio minimo. All'interno di AWS, i livelli di rete vengono definiti principalmente mediante sottoreti, in base agli intervalli di indirizzi IP, all'interno di un Amazon VPC. I livelli possono anche essere definiti utilizzando diversi VPC, ad esempio per raggruppare gli ambienti di microservizi per dominio aziendale. Se utilizzi più VPC, media l'instradamento utilizzando un AWS Transit Gateway. Sebbene ciò fornisca il controllo del traffico a Livello 4 (intervalli di porte e indirizzi IP) utilizzando gruppi di sicurezza e tabella di routing, puoi ottenere un ulteriore controllo utilizzando ulteriori servizi, come AWS PrivateLink, il firewall DNS del risolutore Amazon Route 53, AWS Network Firewall e AWS WAF.

Esamina e fai un inventario di flusso di dati e requisiti di comunicazione dei tuoi carichi di lavoro in termini di parti che avviano la connessione, porte, protocolli e livelli di rete. Valuta i protocolli disponibili per la creazione di connessioni e la trasmissione di dati in modo da selezionare quelli conformi ai tuoi requisiti di protezione (ad esempio, HTTPS anziché HTTP). Acquisisci questi requisiti sia ai limiti delle tue reti sia all'interno di ogni livello. Una volta identificati questi requisiti, esplora le opzioni per consentire il flusso del traffico richiesto solo in ciascun punto di connessione. È bene partire con i gruppi di sicurezza all'interno del VPC, in quanto collegabili a risorse che utilizzano un'interfaccia di rete elastica (ENI), come istanze Amazon EC2, attività Amazon ECS, pod Amazon EKS o database Amazon RDS. A differenza di un firewall Livello 4, un gruppo di sicurezza può avere una regola che consente il traffico da un altro gruppo di sicurezza in base al suo identificatore, riducendo al minimo gli aggiornamenti quando le risorse all'interno del gruppo cambiano nel tempo. Puoi anche filtrare il traffico utilizzando le regole in entrata e in uscita utilizzando i gruppi di sicurezza.

Quando il traffico si sposta tra i VPC, è comune utilizzare il peering VPC per il routing semplice o AWS Transit Gateway per il routing complesso. Questi approcci agevolano i flussi di traffico tra l'intervallo di indirizzi IP delle reti di origine e di destinazione. Tuttavia, se il tuo carico di lavoro richiede solo flussi di traffico tra componenti specifici in diversi VPC, prendi in considerazione l'utilizzo di una connessione punto a punto utilizzando AWS PrivateLink. A tal fine, individua quale servizio dovrebbe agire come produttore e quale dovrebbe agire come consumatore. Implementa un bilanciatore del carico compatibile per il produttore, attiva PrivateLink di conseguenza, quindi accetta una richiesta di connessione da parte del consumatore. Al servizio del produttore viene dunque assegnato un indirizzo IP privato dal VPC del consumatore, utilizzabile dallo stesso per effettuare richieste successive. Questo approccio riduce la necessità di eseguire il peer-to-peer delle reti.

Includi i costi per l'elaborazione dei dati e il bilanciamento del carico come parte della valutazione PrivateLink.

Sebbene i gruppi di sicurezza e PrivateLink agevolino il controllo del flusso tra i componenti dei carichi di lavoro, un'altra considerazione importante riguarda come controllare a quali domini DNS le risorse possono accedere (se presenti). A seconda della configurazione DHCP dei tuoi VPC, puoi prendere in considerazione due diversi servizi AWS a tal scopo. La maggior parte dei consumatori utilizza il servizio DNS predefinito del risolutore Route 53 (chiamato anche server Amazon DNS o AmazonProvidedDNS) disponibile per i VPC all'indirizzo +2 del relativo intervallo CIDR. Con questo approccio, puoi creare regole DNS Firewall e associarle al tuo VPC per determinare quali azioni intraprendere per gli elenchi di domini che fornisci.

Se non stai utilizzando il risolutore Route 53 o se desideri integrare il Resolver con funzionalità di ispezione e controllo del flusso più approfondite oltre al filtro di dominio, prendi in considerazione l'implementazione di un AWS Network Firewall. Questo servizio ispeziona i singoli pacchetti utilizzando regole stateless o stateful per determinare se negare o consentire il traffico. Puoi adottare un approccio simile per filtrare il traffico Web in entrata verso i tuoi endpoint pubblici utilizzando AWS WAF. Per ulteriori indicazioni su questi servizi, consulta SEC05-BP03 Implementazione della protezione basata sulle ispezioni.

Passaggi dell'implementazione

1. Identifica i flussi di dati necessari tra i componenti dei tuoi carichi di lavoro.
2. Applica più controlli con un approccio di difesa approfondita per il traffico in entrata e in uscita, incluso l'uso di gruppi di sicurezza e tabelle di routing.
3. Usa i firewall per definire un controllo granulare sul traffico di rete in entrata, in uscita e attraverso i tuoi VPC, come il firewall DNS del risolutore Route 53, AWS Network Firewall e AWS WAF. Prendi in considerazione l'utilizzo di AWS Firewall Manager per configurare e gestire a livello centrale le regole del firewall in tutta l'organizzazione.

Risorse

Best practice correlate:

- REL03-BP01 Scelta del tipo di segmentazione del carico di lavoro
- SEC09-BP02 Applicazione della crittografia dei dati in transito

Documenti correlati:

- Security best practices for your VPC
- AWS Network Optimization Tips
- Guidance for Network Security on AWS
- Secure your VPC's outbound network traffic in the Cloud AWS

Strumenti correlati:

- AWS Firewall Manager

Video correlati:

- AWS Transit Gateway reference architectures for many VPCs
- Application Acceleration and Protection with Amazon CloudFront, AWS WAF, and AWS Shield
- AWS re:Inforce 2023: Firewalls and where to put them

SEC05-BP03 Implementare la protezione basata sull'ispezione

Imposta i punti di ispezione del traffico tra i livelli di rete per verificare che i dati in transito corrispondano a categorie e schemi previsti. Analizza i flussi di traffico, i metadati e i modelli per identificare, rilevare e rispondere agli eventi in modo più efficace.

Risultato desiderato: ispezione e autorizzazione del traffico che attraversa i livelli di rete. Le decisioni di autorizzazione e rifiuto si basano su regole esplicite, informazioni sulle minacce e deviazioni dai comportamenti di base. Le protezioni diventano più severe man mano che il traffico si avvicina ai dati sensibili.

Anti-pattern comuni:

- Affidamento esclusivo alle regole del firewall basate su porte e protocolli. Mancato sfruttamento di sistemi intelligenti.
- Creazione di regole del firewall basate su specifici modelli di minaccia attuali, soggetti a modifiche.
- Ispezione solo del traffico che transita da una sottorete privata a una pubblica o da una sottorete pubblica a Internet.
- Mancata visione di base del traffico di rete da confrontare per individuare eventuali anomalie di comportamento.

Vantaggi dell'adozione di questa best practice: i sistemi di ispezione ti consentono di creare regole intelligenti, come consentire o negare il traffico solo in presenza di determinate condizioni all'interno dei dati di traffico. Approfitta dei set di regole gestiti AWS e dei partner, basati sulle più recenti informazioni sulle minacce, man mano che il panorama delle minacce cambia nel tempo. In questo modo si riduce l'onere di mantenere le regole e di ricercare gli indicatori di compromissione, riducendo il potenziale di falsi positivi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Ottieni un controllo preciso sul traffico di rete stateful e stateless utilizzando AWS Network Firewall o altri firewall e sistemi di prevenzione delle intrusioni () Marketplace AWS che puoi implementare dietro un Gateway Load Balancer (IPS). GWLBAWS Network Firewall supporta le specifiche open source compatibili con Suricata per proteggere il carico di lavoro. IPS

Sia le soluzioni dei AWS Network Firewall fornitori che utilizzano un GWLB supportano diversi modelli di implementazione dell'ispezione in linea. Ad esempio, è possibile eseguire l'ispezione su VPC base individuale, centralizzarla o implementarla in un modello ibrido in cui il traffico est-ovest attraversa un'ispezione VPC e l'ingresso di Internet viene ispezionato di conseguenza. VPC VPC Un'altra considerazione è se la soluzione supporti l'unwrapping Transport Layer Security (TLS), che consente un'ispezione approfondita dei pacchetti per i flussi di traffico avviati in entrambe le direzioni. Per ulteriori informazioni e dettagli approfonditi su queste configurazioni, consulta la AWS Network Firewall Best Practice guide.

Se utilizzate soluzioni che eseguono out-of-band ispezioni, come l'analisi pcap dei dati a pacchetto provenienti da interfacce di rete che funzionano in modalità promiscua, potete configurare il mirroring del traffico. VPC Il traffico in mirroring viene conteggiato ai fini della larghezza di banda disponibile delle interfacce ed è soggetto agli stessi costi di trasferimento dati del traffico non in mirroring. È possibile verificare se le versioni virtuali di questi dispositivi sono disponibili su Marketplace AWS, che possono supportare la distribuzione in linea dietro a. GWLB

Per i componenti che effettuano transazioni tramite protocolli HTTP basati su protocolli basati, proteggi la tua applicazione dalle minacce comuni con un firewall per applicazioni Web ()WAF. AWS WAFè un firewall per applicazioni Web che ti consente di monitorare e bloccare le richieste HTTP (S) che corrispondono alle tue regole configurabili prima di inviarle ad Amazon API Gateway CloudFront, Amazon AWS AppSync o un Application Load Balancer. Prendi in considerazione l'ispezione approfondita dei pacchetti quando valuti l'implementazione del firewall delle tue applicazioni Web, poiché alcuni richiedono l'interruzione TLS prima dell'ispezione del traffico. Per iniziare AWS WAF,

puoi utilizzare Regole gestite da AWSin combinazione con le tue integrazioni partner o utilizzare le integrazioni dei partner esistenti.

Puoi gestire centralmente AWS WAF AWS Shield Advanced AWS Network Firewall, e i gruppi di VPC sicurezza Amazon in tutta la tua AWS organizzazione con AWS Firewall Manager.

Passaggi dell'implementazione

1. Determina se puoi disciplinare le regole di ispezione in modo ampio, ad esempio attraverso un'ispezioneVPC, o se hai bisogno di un approccio più granulare. VPC
2. Per soluzioni di ispezione in linea:
   a. Se lo utilizzi AWS Network Firewall, crea regole, politiche firewall e il firewall stesso. Una volta configurati questi elementi, puoi indirizzare il traffico verso l'endpoint del firewall per consentire l'ispezione.
   b. Se utilizzi un'appliance di terze parti con un Gateway Load Balancer GWLB (), distribuisci e configura l'appliance in una o più zone di disponibilità. Quindi, crea il tuo servizio endpointGWLB, l'endpoint e configura il routing per il tuo traffico.
3. Per out-of-band le soluzioni di ispezione:
   1. Attiva il mirroring VPC del traffico sulle interfacce in cui è necessario rispecchiare il traffico in entrata e in uscita. Puoi utilizzare EventBridge le regole di Amazon per richiamare una AWS Lambda funzione per attivare il mirroring del traffico sulle interfacce quando vengono create nuove risorse. Indirizza le sessioni di mirroring del traffico al Network Load Balancer davanti all'appliance che elabora il traffico.
4. Per soluzioni di traffico Web in entrata:
   a. Per configurare AWS WAF, inizia configurando una lista di controllo degli accessi Web (web). ACL Il Web ACL è una raccolta di regole con un'azione predefinita (ALLOWoDENY) elaborata in serie che definisce il modo in cui l'utente WAF gestisce il traffico. Puoi creare regole e gruppi personalizzati o utilizzare gruppi di regole AWS gestiti nel tuo WebACL.
   b. Una volta configurato ACL il Web, associalo a una AWS risorsa (come un Application Load Balancer, un API Gateway REST API o una CloudFront distribuzione) per iniziare a proteggere il traffico Web. ACL

Risorse

Documenti correlati:

- What is Traffic Mirroring?

- Implementing inline traffic inspection using third-party security appliances
- AWS Network Firewall architetture di esempio con routing
- Architettura di ispezione centralizzata con AWS Gateway Load Balancer e AWS Transit Gateway

Esempi correlati:

- Best practices for deploying Gateway Load Balancer
- TLSconfigurazione di ispezione per il traffico in uscita crittografato e AWS Network Firewall

Strumenti correlati:

- Marketplace AWS IDS/IPS

SEC05-BP04 Automatizza la protezione della rete

Automatizza l'implementazione delle protezioni di rete utilizzando DevOps pratiche come infrastructure as code (IaC) e pipeline CI/CD. Queste pratiche possono aiutare a tenere traccia delle modifiche apportate alle protezioni di rete attraverso un sistema di controllo delle versioni, a ridurre i tempi di implementazione delle modifiche e a rilevare se le protezioni di rete si allontanano dalla configurazione desiderata.

Risultato desiderato: definizione delle protezioni di rete con modelli e relativo inserimento in un sistema di controllo delle versioni. In caso di nuove modifiche, vengono avviate pipeline automatiche che ne orchestrano test e implementazione. I controlli delle policy e altri test statici sono in atto per convalidare le modifiche prima dell'implementazione. L'implementazione delle modifiche avviene in un ambiente di staging per convalidare il funzionamento previsto dei controlli. Anche l'implementazione negli ambienti di produzione avviene in automatico una volta approvati i controlli.

Anti-pattern comuni:

- Affidamento ai singoli team del carico di lavoro la definizione dell'intero stack di rete, delle protezioni e delle automazioni. Mancata pubblicazione degli aspetti standard dello stack di rete e delle protezioni in modo centralizzato per consentire ai team del carico di lavoro di utilizzarli.
- Affidamento a un team di rete centrale per definire tutti gli aspetti della rete, delle protezioni e delle automazioni. Mancata delega degli aspetti specifici del carico di lavoro dello stack di rete e delle protezioni al team di quel carico di lavoro.

- Individuazione del giusto equilibrio tra centralizzazione e delega tra un team di rete e i team del carico di lavoro, ma mancata applicazione di standard di test e implementazione coerenti nei modelli IaC e nelle pipeline CI/CD. Mancata acquisizione delle configurazioni richieste negli strumenti che controllano l'aderenza dei modelli.

Vantaggi dell'adozione di questa best practice: l'utilizzo di modelli per definire le protezioni di rete consente di tracciare le modifiche e confrontarle nel tempo con un sistema di controllo delle versioni. L'uso dell'automazione per testare e implementare le modifiche crea standardizzazione e prevedibilità, aumentando le possibilità di una corretta implementazione e riducendo le configurazioni manuali ripetitive.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Una serie di controlli di protezione della rete descritti in SEC05-BP02 Controllo dei flussi di traffico all'interno dei livelli di rete e SEC 05-BP03 Implementazione della protezione basata sull'ispezione sono inclusi sistemi di regole gestite che possono essere aggiornati automaticamente in base alle più recenti informazioni sulle minacce. Esempi di protezione degli endpoint Web includono regole gestite e mitigazione automatica a livello di applicazione.AWS WAFAWS Shield Advanced DDoS Utilizza i gruppi di regole AWS Network Firewall gestite per rimanere aggiornato sugli elenchi di domini con scarsa reputazione e sulle firme delle minacce.

Oltre alle regole gestite, ti consigliamo di utilizzare DevOps procedure per automatizzare la distribuzione delle risorse di rete, delle protezioni e delle regole specificate. Puoi acquisire queste definizioni in AWS CloudFormation o in un altro strumento Infrastructure as Code (IaC) di tua scelta, trasferirle in un sistema di controllo delle versioni e implementarle mediante pipeline CI/CD. Utilizzate questo approccio DevOps per ottenere i vantaggi tradizionali della gestione dei controlli di rete, come rilasci più prevedibili, test automatizzati con strumenti come AWS CloudFormation Guarde rilevamento degli scostamenti tra l'ambiente distribuito e la configurazione desiderata.

In base alle decisioni prese nell'ambito di SEC05-BP01 Create network layer, potreste avere un approccio di gestione centralizzato alla creazione VPCs dedicato ai flussi di ingresso, uscita e ispezione. Come descritto nella AWS Security Reference Architecture (AWS SRA), è possibile definirli VPCs in un account dedicato all'infrastruttura di rete. È possibile utilizzare tecniche simili per definire centralmente l'VPCsutilizzo dei carichi di lavoro in altri account, i relativi gruppi di sicurezza, le AWS Network Firewall distribuzioni, le regole di Route 53 Resolver e le configurazioni del DNS firewall e altre risorse di rete. Puoi condividere queste risorse con gli altri tuoi account con AWS

Resource Access Manager. Grazie a questo approccio, puoi semplificare test e implementazione automatici dei controlli di rete nell'account di rete, con una sola destinazione da gestire. Puoi farlo in un modello ibrido, in cui distribuisci e condividi determinati controlli centralmente e deleghi altri controlli ai singoli team del carico di lavoro e ai rispettivi account.

Passaggi dell'implementazione

1. Stabilisci quali aspetti della rete e delle protezioni sono definiti a livello centrale e quali possono essere gestiti dai tuoi team del carico di lavoro.
2. Crea ambienti per testare e implementare le modifiche alla tua rete e alle relative protezioni. Ad esempio, utilizza un account Network Testing e uno Network Production.
3. Determina come archivierai e manterrai i tuoi modelli in un sistema di controllo delle versioni. Archivia i modelli centrali in un repository distinto da quello dei carichi di lavoro, mentre i modelli dei carichi di lavoro possono essere archiviati in repository specifici per quel carico di lavoro.
4. Crea pipeline CI/CD per testare e implementare modelli. Definisci i test per verificare che non ci siano configurazioni errate e che i modelli siano conformi agli standard aziendali.

Risorse

Best practice correlate:

- SEC01-BP06 Automatizza l'implementazione dei controlli di sicurezza standard

Documenti correlati:

- AWS Security Reference Architecture - Network account

Esempi correlati:

- AWS Deployment Pipeline Reference Architecture
- NetDevSecOpsper modernizzare le AWS implementazioni di rete
- Integrazione di test e report AWS CloudFormation di sicurezza AWS Security Hub CSPMAWS CodeBuild

Strumenti correlati:

- AWS CloudFormation

- AWS CloudFormation Guard
- cfn_nag

SEC 6. In che modo proteggi le risorse di calcolo?

Le risorse di calcolo nel carico di lavoro richiedono più livelli di difesa per contribuire alla protezione da minacce esterne e interne. Le risorse di calcolo includono istanze EC2, container, funzioni di AWS Lambda, servizi di database, dispositivi IoT e altro.

Best practice

- SEC06-BP01 Gestione delle vulnerabilità
- SEC06-BP02 Fornitura di dati di calcolo a partire da immagini rinforzate
- SEC06-BP03 Riduzione della gestione manuale e dell'accesso interattivo
- SEC06-BP04 Convalida l'integrità del software
- SEC06-BP05 Automatizzazione della protezione delle risorse di calcolo

SEC06-BP01 Gestione delle vulnerabilità

Scansiona e correggi di frequente le vulnerabilità del codice, delle dipendenze e dell'infrastruttura per proteggerti da nuove minacce.

Risultato desiderato: disponi di una soluzione che analizza continuamente il carico di lavoro alla ricerca di vulnerabilità del software, potenziali difetti ed esposizione involontaria della rete. Hai definito processi e procedure per identificare, assegnare priorità e correggere queste vulnerabilità in base a criteri di valutazione del rischio. Inoltre, hai implementato la gestione automatizzata delle patch per le istanze di calcolo. Il programma di gestione delle vulnerabilità è integrato nel ciclo di vita di sviluppo del software, con soluzioni per la scansione del codice sorgente durante la pipeline CI/CD.

Anti-pattern comuni:

- Assenza di un programma di gestione delle vulnerabilità.
- Esecuzione di patch di sistema senza considerare gravità o prevenzione del rischio.
- Utilizzo di software che ha superato la data di fine vita (EOL) prevista dal fornitore.
- Implementazione del codice in produzione prima di aver analizzato i problemi di sicurezza.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

La gestione delle vulnerabilità è un aspetto fondamentale per mantenere un ambiente cloud sicuro e affidabile. Implica un processo completo che include scansioni di sicurezza, identificazione e definizione delle priorità dei problemi e operazioni di applicazione delle patch per risolvere le vulnerabilità identificate. L'automazione svolge un ruolo fondamentale in questo processo perché facilita la scansione continua dei carichi di lavoro alla ricerca di potenziali problemi ed esposizione involontaria della rete, nonché le operazioni di correzione.

Il modello di responsabilità condivisa di AWS è un concetto fondamentale alla base della gestione delle vulnerabilità. Secondo questo modello, AWS è responsabile della protezione dell'infrastruttura sottostante, compresi hardware, software, reti e strutture in cui vengono eseguiti i servizi AWS. D'altra parte, l'utente è responsabile della protezione dei dati, delle configurazioni di sicurezza e delle attività di gestione associate a servizi come le istanze Amazon EC2 e gli oggetti Amazon S3.

AWS offre una gamma di servizi utili per i programmi di gestione delle vulnerabilità. Amazon Inspector analizza continuamente i carichi di lavoro AWS alla ricerca di vulnerabilità del software e accessi involontari alla rete, mentre Gestione patch di AWS Systems Manager aiuta a gestire l'applicazione delle patch sulle istanze Amazon EC2. Questi servizi possono essere integrati con AWS Security Hub CSPM, un servizio di gestione del livello di sicurezza nel cloud che automatizza i controlli di sicurezza di AWS, centralizza gli avvisi di sicurezza e fornisce una visione completa del livello di sicurezza di un'organizzazione. Inoltre, Sicurezza di Amazon CodeGuru utilizza l'analisi statica del codice per identificare potenziali problemi nelle applicazioni Java e Python durante la fase di sviluppo.

Incorporando pratiche di gestione delle vulnerabilità nel ciclo di vita dello sviluppo software, puoi affrontare in modo proattivo le vulnerabilità prima che vengano introdotte negli ambienti di produzione, riducendo così il rischio di eventi di sicurezza e riducendo al minimo il potenziale impatto delle vulnerabilità.

Passaggi dell'implementazione

1. Comprendi il modello di responsabilità condivisa: consulta il modello di responsabilità condivisa di AWS per comprendere le tue responsabilità in materia di protezione dei carichi di lavoro e dei dati nel cloud. AWS è responsabile della protezione dell'infrastruttura cloud sottostante, mentre tu sei responsabile della protezione delle applicazioni, dei dati e dei servizi che utilizzi.
2. Implementa la scansione delle vulnerabilità: configura un servizio di scansione delle vulnerabilità, come Amazon Inspector, per scansionare automaticamente le istanze di calcolo (ad esempio,

macchine virtuali, container o funzioni serverless) alla ricerca di vulnerabilità software, potenziali difetti ed esposizione involontaria della rete.

3. Stabilisci processi di gestione delle vulnerabilità: definisci processi e procedure per identificare, assegnare priorità e correggere le vulnerabilità. Ciò può includere la pianificazione di scansioni periodiche delle vulnerabilità, la definizione di criteri di valutazione dei rischi e l'individuazione di tempistiche di correzione in base alla gravità della vulnerabilità.
4. Configura la gestione delle patch: utilizza un servizio di gestione delle patch per automatizzare il processo di applicazione delle patch alle istanze di calcolo, sia per i sistemi operativi che per le applicazioni. Puoi configurare il servizio affinché scansioni le istanze alla ricerca di patch mancanti e installi automaticamente le patch in base a una pianificazione. Prendi in considerazione Gestione patch di AWS Systems Manager per fornire questa funzionalità.
5. Configura la protezione contro i malware: implementa meccanismi per rilevare eventuali software dannosi nel tuo ambiente. Ad esempio, puoi utilizzare strumenti come Amazon GuardDuty per analizzare e rilevare eventuali malware, nonché per notificarne la presenza nei volumi EC2 ed EBS. GuardDuty può anche scansionare gli oggetti appena caricati su Amazon S3 alla ricerca di potenziali malware o virus e intervenire per isolarli prima che vengano inseriti nei processi a valle.
6. Integra la scansione delle vulnerabilità nelle pipeline CI/CD: se utilizzi una pipeline CI/CD per l'implementazione delle applicazioni, integra gli strumenti di scansione delle vulnerabilità nella pipeline. Strumenti come Sicurezza di Amazon CodeGuru e le opzioni open source consentono di scansionare il codice sorgente, le dipendenze e gli artefatti alla ricerca di potenziali problemi di sicurezza.
7. Configura un servizio di monitoraggio della sicurezza: configura un servizio di monitoraggio della sicurezza, come AWS Security Hub CSPM, per ottenere una visione completa del tuo livello di sicurezza su più servizi cloud. Il servizio deve raccogliere gli esiti in materia di sicurezza da varie origini e presentarli in un formato standardizzato per facilitare la definizione delle priorità e la correzione.
8. Implementa test di penetrazione delle applicazioni web: se la tua applicazione è un'applicazione web e la tua organizzazione dispone delle competenze necessarie o può usufruire di assistenza esterna, valuta la possibilità di implementare dei test di penetrazione delle applicazioni web per identificare potenziali vulnerabilità nella tua applicazione.
9. Automatizza con l'infrastructure as code: utilizza strumenti di infrastructure as code (IaC), come AWS CloudFormation, per automatizzare l'implementazione e la configurazione delle risorse, inclusi i servizi di sicurezza menzionati in precedenza. Questa pratica consente di creare un'architettura delle risorse più coerente e standardizzata per più account e ambienti.

10. Monitora e migliora continuamente: monitora continuamente l'efficacia del programma di gestione delle vulnerabilità e apporta i miglioramenti necessari. Esamina gli esiti di sicurezza, valuta l'efficacia delle operazioni di correzione e adatta di conseguenza i tuoi processi e strumenti.

Risorse

Documenti correlati:

- AWS Systems Manager
- Panoramica sulla sicurezza di AWS Lambda
- Amazon CodeGuru
- Improved, Automated Vulnerability Management for Cloud Workloads with a New Amazon Inspector
- Automate vulnerability management and remediation in AWS using Amazon Inspector and AWS Systems Manager – Part 1

Video correlati:

- Securing Serverless and Container Services
- Security best practices for the Amazon EC2 instance metadata service

SEC06-BP02 Fornitura di dati di calcolo a partire da immagini rinforzate

Riduci le opportunità di accesso involontario agli ambienti di runtime implementandoli da immagini rafforzate. Acquisisci dipendenze di runtime, come immagini di container e librerie di applicazioni, solo da registri affidabili e verifica le loro firme. Crea i tuoi registri privati per archiviare immagini e librerie attendibili da utilizzare nei tuoi processi di compilazione e implementazione.

Risultato desiderato: l'allocazione delle risorse di calcolo avviene a partire da immagini di base rinforzate. Le dipendenze esterne, ad esempio immagini dei container e librerie di applicazioni, vengono recuperate solo da registri attendibili e ne vengono verificate le firme. Queste sono archiviate in registri privati a cui i processi di compilazione e implementazione possono fare riferimento. Scansiona e aggiorna con regolarità immagini e dipendenze per proteggerti da eventuali vulnerabilità scoperte di recente.

Anti-pattern comuni:

- Acquisizione di immagini e librerie da registri attendibili, ma senza verificarne la firma o eseguire scansioni delle vulnerabilità prima di metterle in uso.
- Rafforzamento delle immagini, ma senza test regolari per individuare nuove vulnerabilità o aggiornarle alla versione più recente.
- Installazione o non rimozione di pacchetti software non necessari durante il ciclo di vita previsto dell'immagine.
- Affidamento esclusivo alle patch per mantenere aggiornate le risorse di calcolo di produzione. La sola applicazione di patch può comunque far sì che nel tempo le risorse di calcolo si allontanino dallo standard rafforzato. L'applicazione delle patch può inoltre non essere in grado di rimuovere le minacce informatiche che un attore pericoloso potrebbero aver installato durante un evento di sicurezza.

Vantaggi dell'adozione di questa best practice: il rafforzamento delle immagini favorisce la riduzione del numero di percorsi disponibili nell'ambiente di runtime, che possono consentire l'accesso non intenzionale a utenti o servizi non autorizzati. Inoltre, può ridurre l'ambito dell'impatto in caso di accesso involontario.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per rafforzare i tuoi sistemi, occorre partire dalle versioni più recenti dei sistemi operativi, delle immagini dei container e delle librerie delle applicazioni. Applica le patch ai problemi noti. Riduci al minimo il sistema rimuovendo applicazioni, servizi, driver dei dispositivi, utenti predefiniti e altre credenziali non necessari. Adotta qualsiasi altra azione necessaria, come la disabilitazione delle porte, per creare un ambiente che disponga solo delle risorse e delle capacità necessarie per i carichi di lavoro. Da questa linea di base è possibile installare software, agenti o altri processi necessari per scopi quali il monitoraggio del carico di lavoro o la gestione delle vulnerabilità.

È possibile ridurre l'onere del rafforzamento dei sistemi utilizzando le linee guida fornite da fonti attendibili, come le guide tecniche per l'implementazione della sicurezza del Center for Internet Security (CIS) e della Defense Information Systems Agency (DISA). STIGs Ti consigliamo di iniziare con una Amazon Machine Image (AMI) pubblicata da AWS o da un APN partner e utilizzare AWS EC2Image Builder per automatizzare la configurazione in base a una combinazione appropriata di CIS controlli e. STIG

Sebbene siano disponibili immagini rinforzate e ricette di EC2 Image Builder che applicano CIS i consigli DISA STIG o, è possibile che la loro configurazione impedisca il corretto funzionamento del

software. In questa situazione, è possibile partire da un'immagine di base non protetta, installare il software e quindi applicare i CIS controlli in modo incrementale per testarne l'impatto. Per qualsiasi CIS controllo che impedisca l'esecuzione del software, verifica se invece riesci a implementare i consigli più dettagliati sulla protezione avanzata in un. DISA Tieni traccia dei diversi CIS controlli e DISA STIG configurazioni che riesci ad applicare con successo. Utilizzateli per definire di conseguenza le vostre ricette di rafforzamento delle EC2 immagini in Image Builder.

Per i carichi di lavoro containerizzati, le immagini rinforzate di Docker sono disponibili nell'archivio pubblico Amazon Elastic Container Registry (). ECR È possibile utilizzare EC2 Image Builder per rafforzare le immagini dei contenitori. AMIs

Analogamente ai sistemi operativi e alle immagini dei contenitori, è possibile ottenere pacchetti di codice (o librerie) da archivi pubblici, tramite strumenti come pip, npm, Maven e. NuGet Ti consigliamo di gestire i pacchetti di codice integrando repository privati, ad esempio all'interno di AWS CodeArtifact, con repository pubblici affidabili. Questa integrazione può gestire il recupero, l'archiviazione e la conservazione dei pacchetti per te. up-to-date I processi di creazione delle applicazioni possono quindi ottenere e testare la versione più recente di questi pacchetti insieme all'applicazione, utilizzando tecniche come Software Composition Analysis (SCA), Static Application Security Testing (SAST) e Dynamic Application Security Testing (DAST).

Per i carichi di lavoro serverless che utilizzano AWS Lambda, semplifica la gestione delle dipendenze dei pacchetti utilizzando i livelli Lambda. Usa i livelli Lambda per configurare un set di dipendenze standard condivise tra diverse funzioni in un archivio autonomo. È possibile creare e gestire i livelli tramite il relativo processo di compilazione, in modo da garantire la permanenza delle funzioni in modo centralizzato. up-to-date

Passaggi dell'implementazione

- Rafforzamento del sistema operativo. Utilizzate immagini di base provenienti da fonti attendibili come base per costruire il vostro hardenedAMIs. Usa EC2Image Builder per personalizzare il software installato sulle tue immagini.
- Rafforzamento delle risorse containerizzate. Configura le risorse containerizzate in modo che rispettino le best practice in materia di sicurezza. Quando utilizzi i contenitori, implementa la scansione delle ECR immagini nella tua pipeline di creazione e, su base regolare, nel tuo archivio di immagini da cercare CVEs nei contenitori.
- Quando si utilizza l'implementazione serverless con AWS Lambda, utilizza i livelli Lambda per separare il codice delle funzioni dell'applicazione e le librerie dipendenti condivise. Configura la

firma del codice per Lambda così da garantire l'esecuzione del solo codice attendibile nelle funzioni Lambda.

Risorse

Best practice correlate:

- OPS05-BP05 Esegui la gestione delle patch

Video correlati:

- Approfondimento sulla sicurezza AWS Lambda

Esempi correlati:

- STIGCompatibile con la compilazione rapida con Image AMI Builder EC2
- Building better container images
- Using Lambda layers to simplify your development process
- Sviluppa e distribuisci AWS Lambda livelli utilizzando Serverless Framework
- Creazione di una pipeline end-to-end AWS DevSecOps CI/CD con strumenti e software open source SCA SAST DAST

SEC06-BP03 Riduzione della gestione manuale e dell'accesso interattivo

Utilizza l'automazione per eseguire attività di implementazione, configurazione, manutenzione e investigazione, laddove possibile. Quando l'automazione non è disponibile, considera l'accesso manuale alle risorse di calcolo in caso di procedure di emergenza o in ambienti sicuri (sandbox).

Risultato desiderato: acquisizione mediante script programmatici e documenti di automazione (runbook) delle azioni autorizzate sulle tue risorse di calcolo. Questi runbook vengono avviati in automatico, attraverso i sistemi di rilevamento delle modifiche, o manualmente, quando è necessario il giudizio umano. L'accesso diretto alle risorse di calcolo è disponibile solo in situazioni di emergenza, quando l'automazione non è disponibile. Tutte le attività manuali vengono inserite in un log e in un processo di revisione per migliorare in modo continuo le capacità di automazione.

Anti-pattern comuni:

- Accesso interattivo alle istanze Amazon EC2 con protocolli come SSH o RDP.

- Mantenimento degli accessi dei singoli utenti, come `/etc/passwd` o gli utenti locali di Windows.
- Condivisione di una password o chiave privata per accedere a un'istanza tra più utenti.
- Installazione del software e creazione o aggiornamento manuali dei file di configurazione.
- Aggiornamento o applicazione di patch manuale al software.
- Accesso a un'istanza per risolvere i problemi.

Vantaggi dell'adozione di questa best practice: l'esecuzione di azioni automatizzate favorisce la riduzione del rischio operativo legato a modifiche non intenzionali ed errori di configurazione. Abolire l'uso di Secure Shell (SSH) e Remote Desktop Protocol (RDP) per l'accesso interattivo significa ridurre la portata dell'accesso alle risorse di calcolo. In tal modo si elimina un percorso comune per le azioni non autorizzate. Acquisire le attività di gestione delle risorse di calcolo in documenti di automazione e script di programmazione significa definire e sottoporre ad audit l'intero ambito delle attività autorizzate a un livello di dettaglio granulare.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

L'accesso a un'istanza è un approccio classico all'amministrazione del sistema. Dopo aver installato il sistema operativo del server, gli utenti in genere accedono manualmente per configurare il sistema e installare il software desiderato. Nel corso del ciclo di vita del server, gli utenti possono accedere per eseguire aggiornamenti del software, applicare patch, modificare le configurazioni e risolvere i problemi.

L'accesso manuale comporta tuttavia una serie di rischi. Richiede un server che ascolti le richieste, come un servizio SSH o RDP, in grado di fornire un potenziale percorso di accesso non autorizzato. Inoltre, aumenta il rischio di errore umano associato all'esecuzione di operazioni manuali. Le conseguenze possono essere incidenti sul carico di lavoro, danneggiamento o distruzione dei dati o altri problemi di sicurezza. L'accesso umano richiede inoltre protezioni contro la condivisione delle credenziali, creando ulteriori costi di gestione.

Per mitigare questi rischi, è possibile implementare una soluzione di accesso remoto basata su agenti, come AWS Systems Manager. AWS Systems Manager L'agente (SSM Agent) avvia un canale crittografato, pertanto non si avvale dell'ascolto di richieste esterne. Per stabilire questo canale su un endpoint VPC, valuta il ricorso alla configurazione di SSM Agent.

Systems Manager offre un controllo granulare delle modalità di interazione con le istanze gestite. Sei tu a definire le automazioni da eseguire, chi può eseguirle e quando possono essere eseguite.

Systems Manager è in grado di applicare patch, installare software e apportare modifiche alla configurazione senza accesso interattivo all'istanza. Systems Manager può inoltre fornire l'accesso a una shell (interprete di comandi) remota e registrare ogni comando richiamato e il relativo output durante la sessione nei log e in Amazon S3. AWS CloudTrail registra le invocazioni delle API di Systems Manager per l'ispezione.

Passaggi dell'implementazione

1. Installa AWS Systems Manager Agent (SSM Agent) sulle istanze Amazon EC2. Verifica se SSM Agent è incluso e avviato in automatico nell'ambito della configurazione AMI di base.
2. Verifica che i ruoli IAM associati ai profili dei tuoi profili delle istanze EC2 includano la policy IAM gestita di `AmazonSSMManagedInstanceCore`.
3. Disabilita SSH, RDP e altri servizi di accesso remoto in esecuzione sulle tue istanze. Puoi farlo eseguendo script configurati nella sezione dei dati utente dei tuoi modelli di avvio o creando AMI personalizzate con strumenti come EC2 Image Builder.
4. Verifica che le regole di ingresso del gruppo di sicurezza applicabili alle tue istanze EC2 non consentano l'accesso sulla porta 22/tcp (SSH) o sulla porta 3389/tcp (RDP). Implementa il rilevamento e l'invio di avvisi su gruppi di sicurezza non configurati correttamente utilizzando servizi come AWS Config.
5. Definisci automazioni, runbook ed esegui comandi appropriati in Systems Manager. Utilizza le policy IAM per definire chi può eseguire queste azioni e le condizioni in base alle quali sono consentite. Testa in modo approfondito queste automazioni in un ambiente non di produzione. Richiama queste automazioni quando necessario, invece di accedere in modo interattivo all'istanza.
6. Utilizza AWS Systems Manager Session Manager per fornire un accesso interattivo alle istanze, quando necessario. Attiva la creazione di log delle attività di sessione per mantenere un audit trail in Amazon CloudWatch Logs o Amazon S3.

Risorse

Best practice correlate:

- REL08-BP04 Esecuzione dell'implementazione utilizzando un'infrastruttura immutabile

Esempi correlati:

- Replacing SSH access to reduce management and security overhead with AWS Systems Manager

Strumenti correlati:

- AWS Systems Manager

Video correlati:

- Controlling User Session Access to Instances in AWS Systems Manager Session Manager

SEC06-BP04 Convalida l'integrità del software

Utilizza la verifica crittografica per convalidare l'integrità degli artefatti software (comprese le immagini) utilizzati dal tuo carico di lavoro. La firma crittografica del software è una tutela contro le modifiche non autorizzate eseguite negli ambienti di calcolo.

Risultato desiderato: ottenimento di tutti gli artefatti da fonti attendibili. I certificati del sito Web del fornitore sono convalidati. Gli artefatti scaricati vengono verificati a livello crittografico tramite le relative firme. Il tuo software è firmato e verificato a livello crittografico dai tuoi ambienti di elaborazione.

Anti-pattern comuni:

- Affidarsi a siti Web di fornitori attendibili per ottenere artefatti software, ma ignorare gli avvisi di scadenza dei certificati. Download senza confermare la validità dei certificati.
- Convalida dei certificati dei siti Web dei fornitori, ma senza verificare a livello crittografico gli artefatti scaricati da questi siti Web.
- Affidarsi esclusivamente a digest o hash per convalidare l'integrità del software. Gli hash stabiliscono che gli artefatti non sono stati modificati rispetto alla versione originale, ma non ne convalidano l'origine.
- Mancata firma di software, codice o librerie di proprietà, anche se utilizzati solo per le proprie implementazioni.

Vantaggi dell'adozione di questa best practice: la convalida dell'integrità degli artefatti da cui dipende il carico di lavoro consente di prevenire l'ingresso di malware negli ambienti di calcolo. La firma del software aiuta a proteggerti dall'esecuzione non autorizzata nei tuoi ambienti di calcolo. Proteggi la catena di approvvigionamento del software firmando e verificando il codice.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Immagini del sistema operativo, immagini dei container e artefatti del codice sono spesso distribuiti con controlli di integrità disponibili, ad esempio attraverso un digest o un hash. Questi permettono ai client di verificare l'integrità elaborando il proprio hash del payload e verificando che sia uguale a quello pubblicato. Sebbene questi controlli aiutino a verificare l'assenza di manomissioni del payload, non ne convalidano la provenienza dalla fonte originale (la sua provenienza). La verifica della provenienza richiede un certificato rilasciato da un'autorità attendibile per firmare digitalmente l'artefatto.

Se utilizzi un software o artefatti scaricati nel tuo carico di lavoro, controlla se il fornitore offre una chiave pubblica per la verifica della firma digitale. Ecco alcuni esempi di come AWS fornisce una chiave pubblica e le istruzioni di verifica per il software che pubblichiamo:

- EC2Image Builder: verifica la firma del download di installazione AWS TOE
- AWS Systems Manager: verifica della firma dell'agente SSM
- Amazon CloudWatch: verifica della firma del pacco dell' CloudWatch agente

Incorpora la verifica della firma digitale nei processi utilizzati per ottenere e rafforzare le immagini, come discusso in SEC06-BP02 Provision compute from hardened images.

È possibile utilizzare AWS Signer per la gestione della verifica delle firme, nonché del ciclo di vita di firma del codice per il tuo software e i tuoi artefatti. AWS Lambda e Amazon Elastic Container Registry offrono entrambi integrazioni con Signer per verificare le firme di codice e immagini. Utilizzando gli esempi nella sezione Risorse, puoi incorporare Signer nelle tue pipeline di integrazione e distribuzione continua (CI/CD) per automatizzare la verifica delle firme e la firma del tuo codice e delle tue immagini.

Risorse

Documenti correlati:

- Cryptographic Signing for Containers
- Le migliori pratiche per proteggere la pipeline di creazione delle immagini dei container utilizzando AWS Signer
- Annuncio della firma di Container Image con AWS Signer Amazon EKS
- Configurazione della firma del codice per AWS Lambda

- Best practices and advanced patterns for Lambda code signing
- Firma del codice tramite CA AWS Certificate Manager privata e chiavi AWS Key Management Service asimmetriche

Esempi correlati:

- Automatizza la firma del codice Lambda con Amazon e CodeCatalyst AWS Signer
- Firma e convalida OCI degli artefatti con AWS Signer

Strumenti correlati:

- AWS Lambda
- AWS Signer
- AWS Certificate Manager
- AWS Key Management Service
- AWS CodeArtifact

SEC06-BP05 Automatizzazione della protezione delle risorse di calcolo

Automatizza le operazioni di protezione delle risorse di calcolo per ridurre la necessità di intervento umano. Usa la scansione automatica per rilevare potenziali problemi all'interno delle tue risorse di calcolo e rimedia con risposte programmatiche automatiche o operazioni di gestione del parco. Incorpora l'automazione nei tuoi processi CI/CD per implementare carichi di lavoro affidabili con dipendenze aggiornate.

Risultato desiderato: tutte le scansioni e le applicazioni di patch alle risorse di calcolo avvengono per mezzo di sistemi automatizzati. Utilizzi la verifica automatica per controllare che immagini e dipendenze del software provengano da origini attendibili e non siano state manomesse. Il controllo dei carichi di lavoro avviene in automatico per verificare la presenza di dipendenze aggiornate, così come la relativa firma per stabilire l'affidabilità negli ambienti di calcolo AWS. Le correzioni automatiche vengono avviate al rilevamento di risorse non conformi.

Anti-pattern comuni:

- Adozione della pratica dell'infrastruttura immutabile, senza però disporre di una soluzione di patch di emergenza o di sostituzione dei sistemi di produzione.

- Utilizzo dell'automazione per correggere le risorse non correttamente configurate, ma senza un meccanismo di annullamento manuale. Possono verificarsi situazioni in cui è necessario modificare i requisiti e sospendere le automazioni fino a quando non si modificano.

Vantaggi dell'adozione di questa best practice: riduzione del rischio di accessi alle risorse di calcolo e relativi utilizzi non autorizzati mediante l'automazione. Contribuisce a evitare che le configurazioni errate si diffondano negli ambienti di produzione e a rilevare e correggere tali configurazioni nel caso in cui si verifichino. L'automazione aiuta anche a rilevare l'accesso non autorizzato delle risorse di calcolo e il loro utilizzo, riducendo i tempi di risposta. In questo modo è possibile ridurre la portata complessiva dell'impatto del problema.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

È possibile applicare le automazioni descritte nelle pratiche del pilastro della sicurezza per proteggere le risorse di calcolo. SEC06-BP01 Gestione delle vulnerabilità illustra come utilizzare Amazon Inspector nelle pipeline CI/CD e per la scansione continua degli ambienti di runtime alla ricerca di vulnerabilità ed esposizioni comuni (CVE) note. Puoi utilizzare AWS Systems Manager per applicare patch o eseguire nuove implementazioni da nuove immagini tramite runbook automatizzati in modo da mantenere il tuo parco di calcolo aggiornato con software e librerie più recenti. Utilizza queste tecniche per ridurre la necessità di processi manuali e l'accesso interattivo alle tue risorse di elaborazione. Consulta SEC06-BP03 Riduzione della gestione manuale e dell'accesso interattivo per scoprire di più.

L'automazione contribuisce anche all'implementazione di carichi di lavoro affidabili, come illustrato in SEC06-BP02 Provisioning di calcolo da immagini rafforzate e SEC06-BP04 Convalida dell'integrità del software. Puoi utilizzare servizi come EC2 Image Builder, AWS Signer, AWS CodeArtifact e Amazon Elastic Container Registry (ECR) per scaricare, verificare, costruire e archiviare immagini e dipendenze di codice rafforzate e approvate. Oltre a Inspector, ognuno di questi può svolgere un ruolo nel processo CI/CD, in modo che il carico di lavoro arrivi in produzione solo quando è confermato che le sue dipendenze sono aggiornate e provengono da origini affidabili. Il carico di lavoro è inoltre firmato in modo che gli ambienti di calcolo AWS, come AWS Lambda e Amazon Elastic Kubernetes Service (EKS), possano verificare l'assenza di manomissioni prima di consentirne l'esecuzione.

Oltre a questi controlli preventivi, è possibile utilizzare l'automazione nei controlli investigativi anche per le risorse di calcolo. Ad esempio, AWS Security Hub CSPM offre lo standard NIST 800-53

Rev. 5 che include controlli come le [EC2.8] EC2 le istanze devono utilizzare Instance Metadata Service versione 2 (IMDSv2).. IMDSv2 utilizza le tecniche di autenticazione della sessione, il blocco delle richieste che contengono un'intestazione X-Forwarded-For HTTP e un TTL di rete pari a 1 per bloccare il traffico proveniente da origini esterne al fine di recuperare informazioni sull'istanza EC2. Questo controllo in Security Hub CSPM può rilevare quando le istanze EC2 utilizzano IMDSv1 e avviare una riparazione automatizzata. Scopri di più su rilevamento e riparazioni automatiche in SEC04-BP04 Avvio della riparazione delle risorse non conformi.

Passaggi dell'implementazione

1. Automatizza la creazione di AMI rafforzate, conformi e consolidate con EC2 Image Builder. È possibile produrre immagini che incorporano i controlli dei benchmark del Center for Internet Security (CIS) o gli standard della Security Technical Implementation Guide (STIG) dalle immagini di base di AWS e dei partner APN.
2. Automatizza la gestione delle configurazioni. Applica e convalida in automatico le configurazioni sicure nelle risorse di calcolo utilizzando un servizio o uno strumento di gestione della configurazione.
    a. Gestione automatizzata della configurazione tramite AWS Config
    b. Gestione automatizzata del livello di sicurezza e conformità tramite AWS Security Hub CSPM
3. Automatizza applicazione delle patch o sostituzione delle istanze Amazon Elastic Compute Cloud (Amazon EC2). AWS Gestione patch di Systems Manager automatizza il processo di applicazione di patch alle istanze gestite con aggiornamenti correlati alla sicurezza e di altro tipo. Gestione patch consente di applicare patch sia per i sistemi operativi sia per le applicazioni
    a. AWS Systems Manager Patch Manager
4. Automatizza la scansione delle risorse di calcolo alla ricerca di vulnerabilità ed esposizioni comuni (CVE) e integra le soluzioni di scansione della sicurezza nella tua pipeline di compilazione.
    a. Amazon Inspector
    b. Scansione delle immagini ECR
5. Prendi in considerazione Amazon GuardDuty per il rilevamento automatico di malware e minacce al fine di proteggere le risorse di calcolo. GuardDuty può anche identificare potenziali problemi in caso di richiamo di una funzione AWS Lambda nel tuo ambiente AWS.
    a. Amazon GuardDuty
6. Prendi in considerazione le soluzioni dei partner AWS. AWS I partner offrono prodotti leader nel settore che sono equivalenti, identici o si integrano ai controlli esistenti negli ambienti on-premises.

Questi prodotti integrano i servizi AWS esistenti per permettere di implementare un'architettura di sicurezza completa e un'esperienza più fluida nel cloud e negli ambienti on-premises.

a. Sicurezza dell'infrastruttura

Risorse

Best practice correlate:

- SEC01-BP06 Implementazione automatizzata dei controlli di sicurezza standard

Documenti correlati:

- Get the full benefits of IMDSv2 and disable IMDSv1 across your AWS infrastructure

Video correlati:

- Security best practices for the Amazon EC2 instance metadata service

Protezione dei dati

Questions

- SEC 7. In che modo classifichi i dati?
- SEC 8. Come proteggi i dati a riposo?
- SEC 9. In che modo proteggi i dati in transito?

SEC 7. In che modo classifichi i dati?

La classificazione fornisce un modo per categorizzare i dati in base ai livelli di criticità e sensibilità, in modo da aiutarti a determinare i controlli di protezione e conservazione appropriati.

Best practice

- SEC07-BP01 Comprendere lo schema di classificazione dei dati
- SEC07-BP02 Applicazione di controlli di protezione dei dati in base alla loro sensibilità
- SEC07-BP03 Automazione dell'identificazione e della classificazione
- SEC07-BP04 Definizione della gestione del ciclo di vita dei dati scalabili

SEC07-BP01 Comprendere lo schema di classificazione dei dati

Comprendi la classificazione dei dati elaborati dal tuo carico di lavoro, i requisiti di gestione, i processi aziendali associati, dove sono archiviati i dati e chi è il relativo proprietario. Lo schema di classificazione e gestione dei dati deve tenere conto dei requisiti legali e di conformità applicabili del carico di lavoro e dei controlli dei dati necessari. Comprendere i dati è il primo passo nel percorso della classificazione dei dati.

Risultato desiderato: comprensione e documentazione ottimali dei tipi di dati presenti nel carico di lavoro. Sono in atto controlli adeguati per proteggere i dati sensibili in base alla loro classificazione. Questi controlli regolano considerazioni quali chi è autorizzato ad accedere ai dati e per quale scopo, la posizione di archiviazione dei dati, qual è la policy di crittografia per tali dati e le modalità di gestione delle chiavi di crittografia, il ciclo di vita dei dati e i requisiti di conservazione, i processi di distruzione opportuni, i processi di backup e ripristino in atto, nonché la verifica degli accessi.

Anti-pattern comuni:

- Non si dispone di una policy formale di classificazione dei dati per definire i livelli di sensibilità dei dati e i relativi requisiti di gestione.
- Non si dispone di una corretta consapevolezza dei livelli di sensibilità dei dati all'interno del carico di lavoro e non si acquisiscono queste informazioni nella documentazione dell'architettura e delle operazioni.
- Mancata applicazione di controlli appropriati sui dati in base alla loro sensibilità e ai requisiti, come indicato nella relativa policy di classificazione e trattamento.
- Mancata indicazione di un feedback sui requisiti di classificazione e trattamento dei dati ai proprietari delle policy.

Vantaggi dell'adozione di questa best practice: eliminazione delle ambiguità circa la corretta gestione dei dati nell'ambito del carico di lavoro grazie a questa pratica. L'applicazione di una policy formale che definisca i livelli di sensibilità dei dati nella propria organizzazione e le relative protezioni richieste, può aiutare a rispettare le normative legali e altre attestazioni e certificazioni di sicurezza informatica. I proprietari dei carichi di lavoro possono avere la certezza di sapere dove sono archiviati i dati sensibili e quali controlli di protezione sono in atto. La loro acquisizione nella documentazione aiuta i nuovi membri del team a comprenderli meglio e a gestire i controlli nelle prime fasi del loro mandato. Queste pratiche possono anche aiutare a ridurre i costi, dimensionando in modo corretto i controlli per ogni tipo di dati.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Nella progettazione di un carico di lavoro, si può prendere in considerazione soluzioni per proteggere i dati sensibili in modo intuitivo. Ad esempio, in un'applicazione multi-tenant, è intuitivo considerare i dati di ciascun tenant come sensibili e mettere in atto protezioni in modo da vietare a un tenant l'accesso ai dati di un altro tenant. Allo stesso modo, è possibile progettare in modo intuitivo i controlli di accesso in modo che solo gli amministratori possano modificare i dati, e che gli altri utenti abbiano solo accesso a livello di lettura o non dispongano di alcun accesso.

Definizione e acquisizione di questi livelli di sensibilità dei dati nelle policy, insieme ai relativi requisiti di protezione dei dati, consente di identificare in modo formale la residenza dei dati nel tuo carico di lavoro. È quindi possibile determinare se sono stati predisposti i controlli giusti, se è possibile verificare i controlli e quali sono le risposte adeguate in caso di gestione errata dei dati.

Per capire dove risiedono i dati sensibili all'interno del carico di lavoro, valuta la possibilità di utilizzare un catalogo dati. Un catalogo dati è un database che mappa i dati nell'organizzazione, con la relativa posizione, il livello di sensibilità e i controlli messi in atto per proteggerli. Valuta inoltre la possibilità di utilizzare i tag delle risorse, se disponibili. Ad esempio, puoi applicare un tag con una chiave di tag di `Classification` e un valore di tag di `PHI` per informazioni sanitarie protette (PHI) e un altro tag con una chiave di tag di `Sensitivity` e un valore di tag pari a `High`. È possibile usare servizi come AWS Config per monitorare tali risorse al fine di rilevare eventuali modifiche e inviare avvisi in caso di modifiche tali da renderle non conformi ai requisiti di protezione (come la modifica delle impostazioni di crittografia). È possibile acquisire la definizione standard delle chiavi tag e dei valori accettabili utilizzando le policy di tag, una funzionalità di AWS Organizations. Non è consigliabile che la chiave o il valore dei tag contenga dati privati o sensibili.

Passaggi dell'implementazione

1. Analizza lo schema di classificazione dei dati e i requisiti di protezione della tua organizzazione.
2. Identifica i tipi di dati sensibili elaborati dai tuoi carichi di lavoro.
3. Acquisisci i dati in un catalogo dedicato che offre una vista unica della posizione in cui risiedono i dati nell'organizzazione e del livello di sensibilità dei dati.
4. Prendi in considerazione l'utilizzo di tag a livello di risorse e dati, laddove disponibili, per etichettare i dati con il relativo livello di sensibilità e altri metadati operativi che possono aiutare nel monitoraggio e nella risposta agli incidenti.
   a. Le policy dei tag AWS Organizations consentono di applicare gli standard di etichettatura.

Risorse

Best practice correlate:

- SUS04-BP01 Implementazione di una policy di classificazione dei dati

Documenti correlati:

- Whitepaper sulla classificazione dei dati
- Best Practices for Tagging AWS Resources

Esempi correlati:

- AWS Organizations Tag Policy Syntax and Examples

Strumenti correlati

- Editor di tag AWS

SEC07-BP02 Applicazione di controlli di protezione dei dati in base alla loro sensibilità

Applica controlli di protezione dei dati che forniscano un livello di controllo adeguato a ciascuna classe di dati definita nella tua policy di classificazione. Questa pratica consente di proteggere i dati sensibili dall'accesso e dall'uso non autorizzati, preservandone al contempo disponibilità e utilizzo.

Risultato desiderato: presenza di una policy di classificazione che definisce i vari livelli di sensibilità dei dati nella tua organizzazione. Per ciascuno di questi livelli di sensibilità, disponi di linee guida chiare per servizi e luoghi di archiviazione e movimentazione approvati e per la loro configurazione richiesta. Implementi controlli per ciascun livello in base al livello di protezione richiesto e ai costi associati. Disponi di un sistema di monitoraggio e di avvisi per rilevare la presenza di dati in luoghi non autorizzati, l'elaborazione in ambienti non autorizzati, l'accesso da parte di soggetti non autorizzati o la configurazione di servizi correlati non conformi.

Anti-pattern comuni:

- Applicazione dello stesso livello di controlli di protezione su tutti i dati. Ciò può portare a un eccesso di controlli di sicurezza per i dati a bassa sensibilità o a una protezione insufficiente dei dati altamente sensibili.

- Mancato coinvolgimento delle parti interessate dei team di sicurezza, conformità e business nella definizione dei controlli sulla protezione dei dati.
- Si trascurano le spese generali e i costi operativi associati all'implementazione e al mantenimento dei controlli sulla protezione dei dati.
- Mancata effettuazione di revisioni periodiche del controllo della protezione dei dati per mantenere l'allineamento con le policy di classificazione.
- Assenza di un inventario completo delle posizioni in cui risiedono i dati a riposo e in transito.

Vantaggi dell'adozione di questa best practice: grazie all'allineamento dei controlli al livello di classificazione dei dati, l'organizzazione può investire in livelli di controllo più elevati, laddove necessario. Ciò può includere l'aumento delle risorse per la sicurezza, il monitoraggio, la misurazione, la correzione e la creazione di report. Se è opportuno disporre di meno controlli, è possibile migliorare l'accessibilità e la completezza dei dati per il personale, i clienti o gli utenti. Questo approccio offre alla tua organizzazione la massima flessibilità nell'utilizzo dei dati, pur rispettandone i requisiti di protezione.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

L'implementazione dei controlli di protezione dei dati in base ai loro livelli di sensibilità comporta diverse fasi fondamentali. In promo luogo, identifica i diversi livelli di sensibilità dei dati all'interno dell'architettura del tuo carico di lavoro (ad esempio, pubblico, interno, riservato e limitato) e valuta il luogo in cui memorizzi ed elabori questi dati. Quindi, definisci i limiti di isolamento dei dati in base al loro livello di sensibilità. Ti consigliamo di separare i dati in diversi Account AWS, utilizzando le policy di controllo dei servizi (SCP) per limitare servizi e azioni consentiti per ciascun livello di sensibilità dei dati. In questo modo, puoi creare forti limiti di isolamento e far rispettare il principio del privilegio minimo.

Una volta definiti i limiti di isolamento, implementa i controlli di protezione adeguati in base ai loro livelli di sensibilità. Consulta le best practice per la protezione dei dati a riposo e la protezione dei dati in transito in modo da implementare controlli pertinenti come la crittografia, i controlli di accesso e gli audit. Prendi in considerazione tecniche come la tokenizzazione o l'anonimizzazione per ridurre il livello di sensibilità dei tuoi dati. Semplifica l'applicazione di policy coerenti sui dati in tutta l'azienda con un sistema centralizzato per la tokenizzazione e la de-tokenizzazione.

Monitora e verifica in modo continuo l'efficacia dei controlli implementati. Rivedi e aggiorna con regolarità lo schema di classificazione dei dati, le valutazioni dei rischi e i controlli di protezione

in base all'evoluzione del panorama di dati e minacce dell'organizzazione. Allinea i controlli di protezione dei dati implementati con normative, standard e requisiti legali pertinenti del settore. Inoltre, procedi alla sensibilizzazione e formazione sulla sicurezza per aiutare i dipendenti a comprendere lo schema di classificazione dei dati e le loro responsabilità nella gestione e protezione dei dati sensibili.

Passaggi dell'implementazione

1. Identifica i livelli di classificazione e sensibilità dei dati all'interno del tuo carico di lavoro.
2. Definisci i limiti di isolamento per ciascun livello e determina una strategia di applicazione.
3. Valuta i controlli definiti che regolano accesso, crittografia, verifica, conservazione e altri aspetti richiesti dalla policy di classificazione dei dati.
4. Valuta le opzioni per ridurre il livello di sensibilità dei dati laddove appropriato, ad esempio utilizzando la tokenizzazione o l'anonimizzazione.
5. Verifica i tuoi controlli utilizzando test e monitoraggio automatici delle risorse configurate.

Risorse

Best practice correlate:

- PERF03-BP01 Uso di un archivio dati dedicato che supporta al meglio i requisiti di accesso e archiviazione dei dati
- COST04-BP05 Applicare policy di conservazione dei dati

Documenti correlati:

- Whitepaper sulla classificazione dei dati
- Best practice per la sicurezza, l'identità e la conformità
- Best practice di AWS KMS
- Encryption best practices and features for AWS services

Esempi correlati:

- Building a serverless tokenization solution to mask sensitive data
- How to use tokenization to improve data security and reduce audit scope

Strumenti correlati:

- AWS Key Management Service (AWS KMS)
- AWS CloudHSM
- AWS Organizations

SEC07-BP03 Automazione dell'identificazione e della classificazione

Automatizzare l'identificazione e la classificazione dei dati può aiutarti a implementare i controlli corretti. L'uso dell'automazione per aumentare la determinazione manuale riduce il rischio di errore umano e di esposizione.

Risultato desiderato: possibilità di verificare se sono in atto controlli adeguati in base alla policy di classificazione e gestione. Strumenti e servizi automatizzati ti aiutano a identificare e classificare il livello di sensibilità dei tuoi dati. L'automazione consente inoltre di monitorare in modo continuo gli ambienti in modo da rilevare e inviare avvisi se i dati vengono archiviati o gestiti in modo non autorizzato, così da poter intraprendere rapidamente azioni correttive.

Anti-pattern comuni:

- Affidarsi esclusivamente a processi manuali per l'identificazione e la classificazione dei dati, che possono essere soggetti a errori e richiedere tempi di lavoro lunghi. Questo può portare a una classificazione dei dati inefficiente e incoerente, soprattutto con l'aumento dei volumi di dati.
- Mancata predisposizione di meccanismi per tracciare e gestire le risorse di dati all'interno dell'organizzazione.
- Si trascura la necessità di un monitoraggio e di una classificazione continui dei dati durante i loro spostamenti e le loro trasformazioni all'interno dell'organizzazione.

Vantaggi dell'adozione di questa best practice: l'automazione di identificazione e classificazione dei dati può garantire un'applicazione più coerente e accurata dei controlli di protezione dei dati, così da ridurre il rischio di errore umano. L'automazione può inoltre fornire visibilità in merito ad accesso e movimento dei dati sensibili, così da rilevare le manipolazioni non autorizzate e intraprendere azioni correttive.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Sebbene si ricorra spesso al giudizio umano per classificare i dati durante le fasi iniziali di progettazione di un carico di lavoro, è opportuno considerare la presenza di sistemi che automatizzino l'identificazione e la classificazione dei dati di test come controllo preventivo. Ad esempio, agli sviluppatori può essere fornito uno strumento o un servizio per analizzare i dati rappresentativi e determinarne la sensibilità. All'interno di AWS, puoi caricare set di dati in Amazon S3 ed eseguirne la scansione mediante Amazon Macie, Amazon Comprehend o Amazon Comprehend Medical. Allo stesso modo, considera la scansione dei dati come parte dei test di unità e integrazione per individuare i casi in cui i dati sensibili non sono previsti. Gli avvisi sui dati sensibili in questa fase possono evidenziare le lacune nelle protezioni prima dell'implementazione in produzione. Altre funzionalità, come il rilevamento di dati sensibili in AWS Glue, Amazon SNS e Amazon CloudWatch, consentono inoltre di rilevare informazioni personali e intraprendere azioni di mitigazione. Per qualsiasi strumento o servizio automatizzato, esamina come definisce i dati sensibili e integralo con altre soluzioni umane o automatizzate per colmare eventuali lacune.

Come controllo investigativo, utilizza il monitoraggio continuo degli ambienti per rilevare l'eventuale archiviazione non conforme dei dati sensibili. In questo modo puoi rilevare situazioni come l'emissione di dati sensibili nei file di log o la loro copia in un ambiente di analisi dei dati senza un'adeguata de-identificazione o redazione. I dati archiviati in Amazon S3 possono essere costantemente monitorati per verificare la presenza di dati sensibili grazie ad Amazon Macie.

Passaggi dell'implementazione

1. Esamina lo schema di classificazione dei dati all'interno dell'organizzazione descritto in SEC07-BP01.
   a. Una volta compreso lo schema di classificazione dei dati dell'organizzazione, puoi stabilire processi accurati per l'identificazione e la classificazione automatica in linea con le policy aziendali.
2. Esegui una scansione iniziale degli ambienti per l'identificazione e la classificazione automatica.
   a. Una prima scansione completa dei dati può aiutare a capire la residenza dei dati sensibili nei tuoi ambienti. Qualora una scansione completa non sia inizialmente richiesta o non possa essere completata in anticipo a causa dei costi, valuta l'adeguatezza delle tecniche di campionamento per raggiungere i tuoi risultati. Ad esempio, Amazon Macie può essere configurato per eseguire un'ampia operazione automatizzata di rilevamento dei dati sensibili nei bucket S3. Questa funzionalità utilizza tecniche di campionamento per eseguire in modo efficiente in termini di costi un'analisi preliminare della residenza dei dati. È quindi possibile eseguire un'analisi più approfondita dei bucket S3 utilizzando un processo di rilevamento dei

dati sensibili. Anche altri archivi di dati possono essere esportati su S3 per essere analizzati da Macie.

   b. Stabilisci il controllo degli accessi definito in SEC07-BP02 per le risorse di archiviazione dei dati identificate durante la scansione.

3. Configura scansioni continue dei tuoi ambienti.
   a. La capacità di rilevamento automatizzata dei dati sensibili di Macie consente di eseguire scansioni continue degli ambienti. I bucket S3 noti e autorizzati a memorizzare dati sensibili possono essere esclusi utilizzando un elenco di permessi in Macie.
4. Incorpora l'identificazione e la classificazione nei processi di compilazione e di test.
   a. Identifica gli strumenti utilizzabili dagli sviluppatori per analizzare i dati alla ricerca di sensibilità mentre i carichi di lavoro sono in fase di sviluppo. Utilizza questi strumenti come parte dei test di integrazione per avvisare quando i dati sensibili sono inaspettati e impedire un'ulteriore implementazione.
5. Implementa un sistema o un runbook per intervenire quando i dati sensibili vengono trovati in luoghi non autorizzati.
   a. Limita l'accesso ai dati utilizzando la correzione automatica. Ad esempio, puoi spostare i dati in un bucket S3 con accesso limitato o assegnare un tag all'oggetto se utilizzi il controllo degli accessi basato su attributi (ABAC). Inoltre, valuta la possibilità di mascherare i dati quando vengono rilevati.
   b. Avvisa i team addetti alla protezione dei dati e alla risposta agli incidenti affinché indaghino sulla causa principale dell'incidente. Ogni informazione appresa può aiutare a prevenire incidenti futuri.

Risorse

Documenti correlati:

- AWS Glue: Detect and process sensitive data
- Using managed data identifiers in Amazon SNS
- Amazon CloudWatch Logs: Help protect sensitive log data with masking

Esempi correlati:

- Enabling data classification for Amazon RDS database with Macie
- Detecting sensitive data in DynamoDB with Macie

Strumenti correlati:

- Amazon Macie
- Amazon Comprehend
- Amazon Comprehend Medical
- AWS Glue

SEC07-BP04 Definizione della gestione del ciclo di vita dei dati scalabili

Esamina i requisiti del ciclo di vita dei dati in relazione ai loro diversi livelli di classificazione e gestione. Ciò può includere le modalità di gestione dei dati quando entrano per la prima volta nell'ambiente, il modo in cui i dati si trasformano e le regole per la loro distruzione. Prendi in considerazione fattori come periodi di conservazione, accesso, audit e monitoraggio della provenienza.

Risultato desiderato: classificazione dei dati il più vicino possibile al momento e all'ora dell'importazione. Quando la classificazione dei dati richiede il mascheramento, la tokenizzazione o altri processi che riducono il livello di sensibilità, si eseguono queste azioni il più vicino possibile al punto e al momento dell'importazione.

Elimini i dati in conformità con la policy in uso quando non è più opportuno conservarli, in base alla loro classificazione.

Anti-pattern comuni:

- Implementazione di un approccio unico alla gestione del ciclo di vita dei dati, senza considerare i diversi livelli di sensibilità e i requisiti di accesso.
- Valutazione della gestione del ciclo di vita solo dal punto di vista dei dati utilizzabili o dei dati di cui si esegue il backup, ma non di entrambi.
- Si presume che i dati immessi nel carico di lavoro siano validi, senza stabilirne il valore o la provenienza.
- Affidamento alla durabilità dei dati come sostituti dei backup e della protezione dei dati.
- Mantenimento dei dati oltre la loro utilità e il periodo di conservazione richiesto.

Vantaggi dell'adozione di questa best practice: una strategia di gestione del ciclo di vita dei dati ben definita e scalabile aiuta a mantenere la conformità normativa, migliora la sicurezza dei dati, ottimizza

i costi di archiviazione e consente l'accesso e la condivisione efficienti dei dati mantenendo i controlli opportuni.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I dati all'interno di un carico di lavoro sono spesso dinamici. La forma che assumono quando entrano nell'ambiente del carico di lavoro può essere diversa da quella che assumono quando vengono archiviati o utilizzati nella logica aziendale, nel reporting, nell'analisi o nel machine learning. Inoltre, il valore dei dati può cambiare nel tempo. Alcuni dati sono di natura temporale e perdono valore con il passare del tempo. Considera l'impatto di queste modifiche ai dati sulla valutazione del tuo schema di classificazione dei dati e dei controlli associati. Laddove possibile, utilizza un meccanismo automatizzato del ciclo di vita, come le policy del ciclo di vita di Amazon S3 e Amazon Data Lifecycle Manager, per configurare i processi di scadenza, archiviazione e conservazione dei dati. Per i dati memorizzati in DynamoDB, puoi utilizzare la funzionalità Time To Live (TTL) per definire un timestamp di scadenza elemento per elemento.

Distingui tra i dati disponibili per l'uso e quelli archiviati come backup. Prendi in considerazione l'utilizzo di AWS Backup per automatizzare il backup dei dati tra tutti i servizi AWS. Gli snapshot di Amazon EBS consentono di copiare un volume EBS e archiviarlo utilizzando le funzionalità di S3, tra cui ciclo di vita, protezione dei dati e accesso ai meccanismi di protezione. Due di questi meccanismi sono S3 Object Lock e AWS Backup Vault Lock, in grado di garantire sicurezza e controllo aggiuntivi ai backup. Gestisci una chiara separazione dei compiti e dell'accesso per i backup. Isola i backup a livello di account per mantenere la separazione dall'ambiente interessato durante un evento.

Un altro aspetto della gestione del ciclo di vita consiste nella registrazione della cronologia dei dati mentre avanzano nel carico di lavoro, chiamato tracciamento della provenienza dei dati. In questo modo hai la certezza di conoscere la provenienza dei dati, le trasformazioni effettuate, il proprietario o il processo che ha apportato le modifiche e la data. Questa cronologia è utile per la risoluzione dei problemi e le analisi in caso di potenziali eventi di sicurezza. Ad esempio, puoi creare log sui metadati relativi alle trasformazioni in una tabella Amazon DynamoDB. All'interno di un data lake, puoi conservare copie dei dati trasformati in diversi bucket S3 per ciascuna fase della pipeline di dati. Archivia le informazioni su schema e timestamp in un AWS Glue Data Catalog. Indipendentemente dalla tua soluzione, considera i requisiti degli utenti finali per determinare gli strumenti appropriati di cui hai bisogno per segnalare la provenienza dei tuoi dati. In questo modo potrai determinare come tracciare al meglio la tua provenienza.

Passaggi dell'implementazione

1. Analizza i tipi di dati, i livelli di sensibilità e i requisiti di accesso del carico di lavoro per classificare i dati e definire strategie di gestione del ciclo di vita appropriate.
2. Progetta e implementa policy di conservazione dei dati e processi di distruzione automatizzata in linea con i requisiti legali, normativi e organizzativi.
3. Stabilisci processi e automazione per il monitoraggio continuo, la verifica e l'adeguamento delle strategie, dei controlli e delle policy di gestione del ciclo di vita dei dati in base all'evoluzione dei requisiti del carico di lavoro e delle normative.
    a. Individua eventuali risorse per le quali non è attivata la gestione automatica del ciclo di vita con AWS Config.

Risorse

Best practice correlate:

- COST04-BP05 Applicare policy di conservazione dei dati
- SUS04-BP03 Utilizzo delle policy per gestire il ciclo di vita dei set di dati

Documenti correlati:

- Whitepaper sulla classificazione dei dati
- AWS Blueprint for Ransomware Defense
- DevOps Guidance: Improve traceability with data provenance tracking

Esempi correlati:

- How to protect sensitive data for its entire lifecycle in AWS
- Build data lineage for data lakes using AWS Glue, Amazon Neptune, and Spline

Strumenti correlati:

- AWS Backup
- Amazon Data Lifecycle Manager
- AWS Identity and Access Management Access Analyzer

SEC 8. Come proteggi i dati a riposo?

Proteggi i dati a riposo implementando più controlli per ridurre il rischio di accessi non autorizzati o altri comportamenti impropri.

Best practice

- SEC08-BP01 Implementazione della gestione sicura delle chiavi
- SEC08-BP02 Applicazione della crittografia dei dati a riposo
- SEC08-BP03 Automatizzazione della protezione dei dati a riposo
- SEC08-BP04 Applicazione del controllo degli accessi

SEC08-BP01 Implementazione della gestione sicura delle chiavi

La gestione sicura delle chiavi include l'archiviazione, la rotazione, il controllo degli accessi e il monitoraggio del materiale relativo alla chiave necessario per proteggere i dati a riposo per il carico di lavoro.

Risultato desiderato: disponibilità di un meccanismo di gestione delle chiavi scalabile, ripetibile e automatizzato. Il meccanismo applica l'accesso con privilegio minimo al materiale relativo alle chiavi e offre il giusto equilibrio tra disponibilità, riservatezza e integrità delle chiavi. È possibile monitorare l'accesso alle chiavi e, se è necessaria la rotazione del materiale delle chiavi, tale operazione può essere eseguita tramite un processo automatizzato. L'accesso al materiale delle chiavi da parte di operatori umani non è consentito.

Anti-pattern comuni:

- Accesso umano a materiale relativo alla chiave non crittografato.
- Creazione di algoritmi crittografici personalizzati.
- Autorizzazioni di accesso al materiale relativo alla chiave di accesso troppo ampie.

Vantaggi dell'adozione di questa best practice: predisponendo un meccanismo di gestione delle chiavi sicuro per il tuo carico di lavoro, puoi contribuire a proteggere i contenuti dagli accessi non autorizzati. Inoltre, la crittografia dei dati potrebbe essere prevista da requisiti normativi per la tua organizzazione. Una soluzione efficace di gestione delle chiavi può fornire meccanismi tecnici finalizzati alla protezione del materiale relativo alle chiavi in linea con tali normative.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

La crittografia dei dati a riposo è un controllo di sicurezza fondamentale. Il carico di lavoro necessita di un meccanismo per archiviare e gestire in modo sicuro il materiale relativo alla chiave utilizzato per crittografare i dati a riposo.

AWS offre AWS Key Management Service (AWS KMS) per fornire uno spazio di archiviazione durevole, sicuro e ridondante per le chiavi AWS KMS. Molti servizi AWS si integrano con AWS KMS per supportare la crittografia dei dati. AWS KMS utilizza moduli di sicurezza hardware conformi allo standard FIPS 140-3 di livello 3 per proteggere le chiavi. Non esiste un meccanismo per esportare le chiavi AWS KMS convertendole in testo semplice.

Nell'implementazione di carichi di lavoro mediante una strategia multi-account, le chiavi AWS KMS devono essere conservate nello stesso account del carico di lavoro che le utilizza. In questo modello distribuito la responsabilità della gestione delle chiavi AWS KMS spetta al tuo team. In altri casi d'uso, la tua organizzazione può scegliere di archiviare le chiavi AWS KMS in un account centralizzato. Questa struttura centralizzata richiede policy aggiuntive per consentire l'accesso multi-account richiesto affinché l'account del carico di lavoro possa accedere alle chiavi di accesso archiviate nell'account centralizzato, ma può essere più applicabile nei casi d'uso in cui una singola chiave è condivisa tra Account AWS multipli.

Indipendentemente dalla posizione in cui è archiviato il materiale relativo alla chiave, l'accesso alla chiave deve essere strettamente controllato mediante l'uso di policy della chiave e policy IAM. Le policy della chiave costituiscono la modalità principale per controllare l'accesso a una chiave AWS KMS. Inoltre, AWS KMS garantisce che le chiavi possano fornire l'accesso ai servizi AWS per crittografare e decrittografare i dati per tuo conto. Consulta le linee guida per il controllo degli accessi alle chiavi AWS KMS.

È necessario monitorare l'uso delle chiavi di crittografia per rilevare eventuali modelli di accesso insoliti. Le operazioni eseguite utilizzando chiavi gestite da AWS e chiavi gestite dal cliente archiviate in AWS KMS, possono essere registrate in AWS CloudTrail e devono essere riviste periodicamente. Presta particolare attenzione al monitoraggio degli eventi di eliminazione delle chiavi. Per ridurre le probabilità di distruzione accidentale o dolosa del materiale relativo alla chiave, gli eventi di eliminazione delle chiavi non hanno efficacia immediata. I tentativi di eliminazione delle chiavi in AWS KMS sono soggetti a un periodo di attesa, che per impostazione predefinita è di 30 giorni (con un minimo di 7 giorni), così da garantire agli amministratori il tempo di rivedere queste azioni e annullare la richiesta, se necessario.

La maggior parte dei servizi AWS utilizza AWS KMS secondo una modalità chiara per te: il tuo unico requisito è decidere se utilizzare una chiave gestita da AWS o dal cliente. Se il carico di lavoro richiede l'uso diretto di AWS KMS per crittografare o decrittografare i dati, occorre utilizzare la crittografia a busta per proteggere i tuoi dati. L'SDK di crittografia di AWS è in grado di fornire alle applicazioni primitive la crittografia lato client per implementare la crittografia a busta e integrarle con AWS KMS.

Passaggi dell'implementazione

1. Determina le opzioni di gestione delle chiavi adeguate (gestite da AWS o dal cliente) per la chiave.
    a. Per facilitare l'uso, AWS offre chiavi AWS di proprietà e gestite da AWS per la maggior parte dei servizi, fornendo funzionalità di crittografia a riposo senza la necessità di gestire il materiale o le policy della chiave.
    b. Quando utilizzi chiavi gestite dal cliente, prendi in considerazione il keystore predefinito per fornire il miglior equilibrio tra agilità, sicurezza, sovranità dei dati e disponibilità. Per altri casi d'uso può essere richiesto l'uso di archivi di chiavi personalizzati con AWS CloudHSM o l'archivio chiavi esterno.
2. Consulta l'elenco dei servizi che stai utilizzando per il tuo carico di lavoro per capire come AWS KMS si integra con il servizio. Ad esempio, le istanze EC2 possono utilizzare volumi EBS crittografati; verifica che anche gli snapshot Amazon EBS create da tali volumi siano crittografate utilizzando una chiave gestita dal cliente e mitigando la divulgazione accidentale di dati di snapshot non crittografati.
    a. How AWS services use AWS KMS
    b. Per informazioni dettagliate sulle opzioni di crittografia offerte da un servizio AWS, consulta l'argomento relativo alla crittografia dei dati a riposo nella guida per l'utente o nella guida per gli sviluppatori del servizio.
3. Implementa AWS KMS: AWS KMS semplifica la creazione e la gestione delle chiavi e controlla l'uso della crittografia in un'ampia gamma di servizi AWS e nelle tue applicazioni.
    a. Nozioni di base: AWS Key Management Service (AWS KMS)
    b. Consulta le best practices for access control to your AWS KMS keys.
4. Considera l'SDK di crittografia AWS: utilizza l'SDK di crittografia AWS con l'integrazione di AWS KMS quando la tua applicazione deve crittografare i dati lato client.
    a. SDK di crittografia AWS
5. Abilita IAM Access Analyzer per rivedere e inviare notifiche in automatico se esistono policy della chiave AWS KMS eccessivamente permissive.

   a. Valuta la possibilità di utilizzare controlli delle policy personalizzati per verificare che l'aggiornamento di una policy delle risorse non conceda l'accesso pubblico alle chiavi KMS.

6. Abilita Security Hub CSPM per ricevere notifiche in caso di policy della chiave configurate in modo errato, chiavi programmate per essere eliminate o chiavi senza la rotazione automatica abilitata.
7. Determina il livello di log appropriato per le tue chiavi AWS KMS. Poiché le chiamate a AWS KMS, inclusi gli eventi di sola lettura, vengono registrate, i log CloudTrail associati a AWS KMS possono diventare voluminosi.
   a. Alcune organizzazioni preferiscono la segregazione dell'attività di log di AWS KMS in un percorso separato. Per maggiori dettagli, consulta la sezione Logging AWS KMS API calls with CloudTrail della guida per gli sviluppatori AWS KMS.

Risorse

Documenti correlati:

- AWS Key Management Service
- AWS cryptographic services and tools
- Protezione dei dati Amazon S3 tramite la crittografia
- Crittografia envelope
- Impegno per la sovranità digitale
- Demystifying AWS KMS key operations, bring your own key, custom key store, and ciphertext portability
- AWS Key Management Service Dettagli della crittografia di

Video correlati:

- How Encryption Works in AWS
- Securing Your Block Storage on AWS
- AWS data protection: Using locks, keys, signatures, and certificates

Esempi correlati:

- Implement advanced access control mechanisms using AWS KMS

SEC08-BP02 Applicazione della crittografia dei dati a riposo

Crittografando i dati privati a riposo è possibile mantenere la riservatezza e fornire un ulteriore livello di protezione contro la divulgazione o esfiltrazione involontaria dei dati. La crittografia protegge i dati in modo che non possano essere letti o consultati senza prima essere stati decrittografati. Effettua un inventario e un controllo dei dati non crittografati per mitigare i rischi associati all'esposizione dei dati.

Risultato desiderato: disponi di meccanismi che effettuano la crittografia dei dati privati per impostazione predefinita quando sono a riposo. Questi meccanismi aiutano a mantenere la riservatezza dei dati e forniscono un ulteriore livello di protezione contro la divulgazione o esfiltrazione involontaria dei dati. Mantieni un inventario dei dati non crittografati e comprendi i controlli in atto per proteggerli.

Anti-pattern comuni:

- Mancato utilizzo di configurazioni con crittografia predefinita.
- Accesso estremamente permissivo alle chiavi di decrittografia.
- Mancato monitoraggio dell'uso delle chiavi di crittografia e decrittografia.
- Memorizzazione di dati non crittografati.
- Utilizzo della stessa chiave di crittografia per tutti i dati, indipendentemente dall'uso, dal tipo e dalla classificazione dei dati stessi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Mappa le chiavi di crittografia in base alle classificazioni dei dati all'interno dei carichi di lavoro. Questo approccio favorisce la protezione dei dati da accessi eccessivamente permissivi in caso di utilizzo di una sola chiave di crittografia o di un numero molto ridotto di chiavi di crittografia (vedi SEC07-BP01 Comprendere lo schema di classificazione dei dati).

AWS Key Management Service (AWS KMS) si integra con molti servizi AWS per semplificare la crittografia dei dati a riposo. Ad esempio, in Amazon Elastic Compute Cloud (Amazon EC2) puoi impostare la crittografia predefinita sugli account in modo che i nuovi volumi EBS vengano crittografati in automatico. Quando utilizzi AWS KMS, devi considerare il livello di restrizione dei dati. Le chiavi AWS KMS predefinite e controllate dal servizio sono gestite e utilizzate da AWS per tuo conto. Per i dati sensibili che richiedono un accesso granulare alla chiave di crittografia sottostante, è opportuno considerare le chiavi gestite dal cliente (CMK). L'utente ha il pieno controllo sulle CMK,

anche per quanto riguarda la rotazione e la gestione degli accessi attraverso l'uso di policy sulla chiave.

Inoltre, servizi come Amazon Simple Storage Service (Amazon S3) effettuano ora la crittografia di tutti i nuovi oggetti per impostazione predefinita. Questa implementazione offre una maggiore sicurezza senza alcun impatto sulle prestazioni.

Altri servizi, ad esempio Amazon Elastic Compute Cloud (Amazon EC2) o Amazon Elastic File System (Amazon EFS), supportano impostazioni per la crittografia predefinita. Puoi utilizzare anche Regole di AWS Config per verificare in automatico che sia in uso la crittografia per i volumi Amazon Elastic Block Store (Amazon EBS), le istanze Amazon Relational Database Service (Amazon RDS), i bucket Amazon S3 e altri servizi all'interno della tua organizzazione.

AWS offre anche soluzioni per la crittografia lato client, consentendo di crittografare i dati prima di caricarli nel cloud. AWS Encryption SDK offre un modo per la crittografia dei dati mediante la crittografia a busta. L'utente fornisce la chiave di wrapping e AWS Encryption SDK genera una chiave dati unica per ogni oggetto di dati che crittografa. Prendi in considerazione AWS CloudHSM se hai bisogno di un modulo di sicurezza hardware (HSM) gestito single-tenant. AWS CloudHSM consente di generare, importare e gestire le chiavi crittografiche su un HSM convalidato FIPS 140-2 di livello 3. Alcuni casi d'uso di AWS CloudHSM includono la protezione delle chiavi private per il rilascio di un'autorità di certificazione (CA) e l'abilitazione della crittografia dei dati trasparente (TDE) per i database Oracle. Il client SDK AWS CloudHSM fornisce un software che consente di crittografare i dati sul lato client utilizzando le chiavi memorizzate all'interno di AWS CloudHSM prima di caricare i dati in AWS. La crittografia lato client Amazon DynamoDB consente inoltre di crittografare e firmare gli elementi prima del caricamento in una tabella DynamoDB.

Passaggi dell'implementazione

- Configura la crittografia predefinita per nuovi volumi Amazon EBS: specifica che desideri che tutti i volumi Amazon EBS appena creati vengano creati in forma crittografata, con la possibilità di utilizzare la chiave predefinita fornita da AWS oppure una chiave creata da te.
- Configura Amazon Machine Image (AMI) crittografate: copiando un'AMI esistente con crittografia abilitata, verrà eseguita la crittografia automatica di volumi root e snapshot.
- Configura la crittografia Amazon RDS: configura la crittografia per cluster e snapshot del database Amazon RDS a riposo abilitando l'opzione di crittografia.
- Crea e configura chiavi AWS KMS con policy che limitano l'accesso ai principali opportuni per ciascuna classificazione dei dati: ad esempio, crea una chiave AWS KMS per la crittografia dei dati di produzione e una chiave diversa per quella dei dati di sviluppo o di test. Puoi anche fornire

l'accesso alle chiavi ad altri Account AWS. Considera la possibilità di predisporre account diversi per gli ambienti di sviluppo e di produzione. Qualora il tuo ambiente di produzione richieda la decodifica degli artefatti nell'account di sviluppo, puoi modificare la policy CMK utilizzata in modo da crittografare gli artefatti di sviluppo per consentire all'account di produzione di decrittografare tali artefatti. L'ambiente di produzione può quindi importare i dati decrittografati per utilizzarli nella produzione.

- Configura la crittografia nei servizi AWS aggiuntivi: per gli altri servizi AWS che utilizzi, consulta la documentazione di sicurezza relativa al servizio interessato per determinare le opzioni di crittografia del servizio.

Risorse

Documenti correlati:

- AWS Crypto Tools
- AWS Encryption SDK
- AWS KMS Cryptographic Details Whitepaper
- AWS Key Management Service
- AWS cryptographic services and tools
- Amazon EBS Encryption
- Default encryption for Amazon EBS volumes
- Encrypting Amazon RDS Resources
- Come si attiva la crittografia predefinita per un bucket Amazon S3?
- Protezione dei dati Amazon S3 tramite la crittografia

Video correlati:

- How Encryption Works in AWS
- Securing Your Block Storage on AWS

SEC08-BP03 Automatizzazione della protezione dei dati a riposo

Usa l'automazione per convalidare e applicare i controlli dei dati a riposo. Usa la scansione automatica per rilevare le configurazioni errate delle soluzioni di archiviazione di dati ed esegui le correzioni attraverso la risposta programmatica automatica, ove possibile. Incorpora l'automazione

nei tuoi processi CI/CD per rilevare le configurazioni errate dell'archiviazione di dati prima che vengano implementate in produzione.

Risultato desiderato: scansione e monitoraggio da parte di sistemi automatizzati delle posizioni di archiviazione di dati per individuare configurazioni errate dei controlli, accessi non autorizzati e usi imprevisti. Il rilevamento delle posizioni di archiviazione non configurate avvia correzioni automatiche. I processi automatizzati creano backup dei dati e archiviano copie immutabili al di fuori dell'ambiente originale.

Anti-pattern comuni:

- Mancata tenuta in considerazione delle opzioni per abilitare la crittografia dalle impostazioni predefinite, ove supportate.
- Mancata tenuta in considerazione degli eventi di sicurezza, oltre a quelli operativi, quando si formula una strategia di backup e ripristino automatizzata.
- Mancata applicazione delle impostazioni di accesso pubblico per i servizi di archiviazione.
- Assenza di monitoraggio e audit dei controlli per proteggere i dati a riposo.

Vantaggi dell'adozione di questa best practice: prevenzione grazie all'automazione del rischio di configurazioni errate delle posizioni di archiviazione di dati e dell'ingresso di configurazioni errate negli ambienti di produzione. Questa best practice aiuta anche a rilevare e correggere eventuali configurazioni errate.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

L'automazione è un tema ricorrente in tutte le pratiche per la protezione dei dati a riposo. SEC01-BP06 Implementazione automatizzata dei controlli di sicurezza standard illustra come acquisire la configurazione delle risorse utilizzando modelli di infrastructure as code (IaC), ad esempio con AWS CloudFormation. Questi modelli sono vincolati a un sistema di controllo della versione e consentono di distribuire risorse su AWS tramite una pipeline CI/CD. Queste tecniche si applicano anche all'automazione della configurazione delle soluzioni di archiviazione di dati, come le impostazioni di crittografia sui bucket Amazon S3.

Puoi controllare le impostazioni che definisci nei tuoi modelli IaC per eventuali configurazioni errate nelle pipeline CI/CD utilizzando le regole in AWS CloudFormation Guard. Puoi monitorare impostazioni non ancora disponibili in CloudFormation o in altri strumenti IaC per evitare configurazioni errate con AWS Config. È possibile correggere in automatico gli avvisi generati da

Config per configurazioni errate, come illustrato in SEC04-BP04 Avvio della riparazione delle risorse non conformi.

L'utilizzo dell'automazione come parte della strategia di gestione delle autorizzazioni è anche parte integrante delle protezioni automatizzate dei dati. SEC03-BP02 Concessione dell'accesso con privilegio minimo e SEC03-BP04 Riduzione delle autorizzazioni in modo continuo illustrano la configurazione delle policy di accesso con privilegio minimo monitorate di continuo da AWS Identity and Access Management Access Analyzer per generare esiti quando è possibile ridurre le autorizzazioni. Oltre all'automazione per il monitoraggio delle autorizzazioni, puoi configurare Amazon GuardDuty in modo da rilevare comportamenti anomali di accesso ai dati per i tuoi volumi EBS (tramite un'istanza EC2), bucket S3 e i database di Amazon Relational Database Service.

L'automazione svolge rileva inoltre i casi di archiviazione di dati sensibili in luoghi non autorizzati. SEC07-BP03 Automazione dell'identificazione e della classificazione illustra in che modo Amazon Macie può monitorare i bucket S3 alla ricerca di dati sensibili imprevisti e generare avvisi in grado di avviare una risposta automatica.

Segui le pratiche di REL09 In che modo eseguire il backup dei dati? per sviluppare una strategia automatizzata di backup e ripristino dei dati. Il backup e il ripristino dei dati sono importanti tanto per il ripristino da eventi di sicurezza quanto per gli eventi operativi.

Passaggi dell'implementazione

1. Acquisisci la configurazione dell'archiviazione di dati nei modelli IaC. Utilizza i controlli automatizzati nelle pipeline CI/CD per rilevare configurazioni errate.
   a. Puoi utilizzare CloudFormation per i modelli IaC e CloudFormation Guard per verificare la presenza di errori di configurazione nei modelli.
   b. Utilizza AWS Config per eseguire le regole in modalità di valutazione proattiva. Utilizza questa impostazione per verificare la conformità di una risorsa come passaggio della pipeline CI/CD prima di crearla.
2. Monitora le risorse per individuare eventuali configurazioni errate dell'archiviazione di dati.
   a. Imposta AWS Config in modo che monitori le risorse di archiviazione di dati al fine di rilevare eventuali modifiche nelle configurazioni di controllo e generare avvisi per richiamare correzioni in caso di rilevamento di una configurazione errata.
   b. Consulta SEC04-BP04 Avvio della riparazione delle risorse non conformi per ulteriori indicazioni sulle correzioni automatiche.
3. Monitora e riduci in modo continuo le autorizzazioni di accesso ai dati tramite l'automazione.

   a. È possibile eseguire IAM Access Analyzer in modo continuo così da generare avvisi in caso di potenziale riduzione delle autorizzazioni.
4. Monitora e avvisa in caso di comportamenti anomali di accesso ai dati.
   a. GuardDuty analizza sia le firme note delle minacce sia le deviazioni dai comportamenti di accesso di base per le risorse di archiviazione di dati, come volumi EBS, bucket S3 e database RDS.
5. Monitora e invia avvisi sui dati sensibili archiviati in luoghi inaspettati.
   a. Usa Amazon Macie per una scansione continua dei tuoi bucket S3 alla ricerca di dati sensibili.
6. Automatizza i backup sicuri e crittografati dei tuoi dati.
   a. AWS Backup è un servizio gestito che permette di creare backup di varie origini dati su AWS. Elastic Disaster Recovery ti consente di copiare carichi di lavoro completi del server e mantenere una protezione continua dei dati con un obiettivo del punto di ripristino (RPO) misurato in secondi. È possibile configurare entrambi i servizi in modo che lavorino all'unisono per automatizzare la creazione di backup dei dati e la loro copia in posizioni di failover. Questo può aiutare a mantenere i dati disponibili in caso di eventi operativi o di sicurezza.

Risorse

Best practice correlate:

- SEC01-BP06 Implementazione automatizzata dei controlli di sicurezza standard
- SEC03-BP02 Concessione dell'accesso con privilegio minimo
- SEC03-BP04 Riduzione delle autorizzazioni in modo continuo
- SEC04-BP04 Avvio della riparazione delle risorse non conformi
- SEC07-BP03 Automazione dell'identificazione e della classificazione
- REL09-BP02 Protezione e crittografia dei backup
- REL09-BP03 Esecuzione del backup dei dati in automatico

Documenti correlati:

- AWS Prescriptive Guidance: Automatically encrypt existing and new Amazon EBS volumes
- Ransomware Risk Management on AWS Using the NIST Cyber Security Framework (CSF)

Esempi correlati:

- How to use AWS Config proactive rules and AWS CloudFormation Hooks to prevent creation of noncompliant cloud resources
- Automate and centrally manage data protection for Amazon S3 with AWS Backup
- AWS re:Invent 2023 - Implement proactive data protection using Amazon EBS snapshots
- AWS re:Invent 2022 - Build and automate for resilience with modern data protection

Strumenti correlati:

- AWS CloudFormation Guard
- AWS CloudFormation Guard Rules Registry
- IAM Access Analyzer
- Amazon Macie
- AWS Backup
- Elastic Disaster Recovery

SEC08-BP04 Applicazione del controllo degli accessi

Per proteggere i dati a riposo, applica il controllo degli accessi utilizzando meccanismi come l'isolamento e il controllo delle versioni. Applica i controlli in base al privilegio minimo e all'accesso condizionale. Impedisci che venga consentito l'accesso pubblico ai dati.

Risultato desiderato: puoi verificare che l'accesso ai dati sia consentito solo agli utenti autorizzati, in base alle necessità. La protezione dei dati è assicurata da backup regolari e dal controllo delle versioni, per evitare che la modifica dei dati o la loro eliminazione intenzionale o non voluta. L'isolamento dei dati critici dagli altri dati ne protegge la riservatezza e l'integrità.

Anti-pattern comuni:

- Archiviazione dei dati con requisiti di sensibilità o classificazione diversi.
- Utilizzo di autorizzazioni troppo permissive sulle chiavi di decrittografia.
- Classificazione impropria dei dati.
- Nessun mantenimento di backup dettagliati dei dati importanti.
- Accesso persistente ai dati di produzione.
- Nessun audit dell'accesso ai dati o revisione periodica delle autorizzazioni.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

La protezione dei dati a riposo è importante per mantenere l'integrità, la riservatezza e la conformità dei dati ai requisiti normativi. Per ottenere tale risultato puoi implementare più controlli, inclusi controllo degli accessi, isolamento, accesso condizionale e controllo delle versioni.

Puoi applicare il controllo degli accessi con il principio del privilegio minimo, che fornisce solo le autorizzazioni necessarie agli utenti e ai servizi per eseguire le varie attività. È incluso l'accesso alle chiavi di crittografia. Rivedi le policy AWS Key Management Service (AWS KMS) per verificare che il livello di accesso concesso sia appropriato e che si applichino le condizioni pertinenti.

Puoi separare i dati in base a diversi livelli di classificazione utilizzando Account AWS distinti per ogni livello e gestire tali account con AWS Organizations. Tale isolamento può aiutare a prevenire l'accesso non autorizzato e a ridurre al minimo il rischio di esposizione dei dati.

Rivedi periodicamente il livello di accesso concesso nelle policy dei bucket Amazon S3. Evita di utilizzare bucket leggibili o scrivibili pubblicamente a meno che ciò non sia assolutamente necessario. Valuta la possibilità di utilizzare AWS Config per rilevare i bucket disponibili pubblicamente e Amazon CloudFront per distribuire i contenuti da Amazon S3. Verifica che i bucket che non devono consentire l'accesso pubblico siano configurati correttamente a tale scopo.

Implementa meccanismi di controllo delle versioni e Object Lock per i dati critici archiviati in Amazon S3. Il controllo delle versioni di Amazon S3 preserva le versioni precedenti degli oggetti per recuperare i dati in caso di cancellazioni o sovrascritture accidentali. Amazon S3 Object Lock fornisce un controllo degli accessi obbligatorio per gli oggetti, che impedisce che questi ultimi vengano eliminati o sovrascritti, anche dall'utente root, fino alla scadenza del blocco. Inoltre, Amazon Glacier Vault Lock offre una funzionalità simile per gli archivi memorizzati in Amazon Glacier.

Passaggi dell'implementazione

1. Implementa il controllo degli accessi con il principio del privilegio minimo:
   - Verifica le autorizzazioni di accesso concesse a utenti e servizi e verifica che dispongano solo delle autorizzazioni necessarie per svolgere le rispettive attività.
   - Verifica l'accesso alle chiavi di crittografia controllando le policy AWS Key Management Service (AWS KMS).
2. Separa i dati in base a diversi livelli di classificazione:

- Utilizza Account AWS distinti per ogni livello di classificazione dei dati.
- Gestisci questi account utilizzando AWS Organizations.

3. Verifica le autorizzazioni per bucket e oggetti Amazon S3:
   - Rivedi periodicamente il livello di accesso concesso nelle policy dei bucket Amazon S3.
   - Evita di utilizzare bucket leggibili o scrivibili pubblicamente a meno che ciò non sia assolutamente necessario.
   - Valuta la possibilità di utilizzare AWS Config per rilevare i bucket pubblicamente disponibili.
   - Utilizza Amazon CloudFront per distribuire contenuti da Amazon S3.
   - Verifica che i bucket che non devono consentire l'accesso pubblico siano configurati correttamente a tale scopo.
   - Puoi applicare lo stesso processo di revisione per i database e qualsiasi altra origine dati che utilizzi l'autenticazione IAM, come SQS o datastore di terze parti.
4. Utilizza il Sistema di analisi degli accessi AWS IAM:
   - Puoi configurare AWS IAM Access Analyzer per analizzare i bucket Amazon S3 e generare esiti quando una policy S3 concede l'accesso a un'entità esterna.
5. Implementa meccanismi di controllo delle versioni e Object Lock:
   - Utilizza il controllo delle versioni di Amazon S3 per preservare le versioni precedenti degli oggetti, consentendo così il ripristino in caso di cancellazioni o sovrascritture accidentali.
   - Utilizza Amazon S3 Object Lock per fornire un controllo degli accessi obbligatorio per gli oggetti, che impedisce che questi ultimi vengano eliminati o sovrascritti, anche dall'utente root, fino alla scadenza del blocco.
   - Utilizza Amazon Glacier Vault Lock per gli archivi in Amazon Glacier.
6. Utilizza l'Inventario Amazon S:
   - Puoi utilizzare l'Inventario Amazon S3 per eseguire audit e segnalare lo stato di replica e crittografia dei tuoi oggetti S3.
7. Verifica le autorizzazioni di condivisione di Amazon EBS e AMI:
   - Esamina le tue autorizzazioni di condivisione di Amazon EBS e AMI per verificare che le immagini e i volumi non vengano condivisi con Account AWS esterni al tuo carico di lavoro.
8. Rivedi periodicamente le condivisioni di AWS Resource Access Manager:
   - Puoi utilizzare AWS Resource Access Manager per condividere risorse come le policy AWS Network Firewall, le regole del risolutore Amazon Route 53 e le sottoreti all'interno dei tuoi Amazon VPC.

- Sottoponi regolarmente ad audit le risorse condivise e interrompi la condivisione delle risorse che non devono più essere condivise.

Risorse

Best practice correlate:

- SEC03-BP01 Definizione dei requisiti di accesso
- SEC03-BP02 Concessione dell'accesso con privilegio minimo

Documenti correlati:

- AWS KMS Cryptographic Details Whitepaper
- Introduzione alla gestione delle autorizzazioni di accesso alle risorse di Amazon S
- Panoramica della gestione dell'accesso alle risorse AWS KMS
- Regole di AWS Config
- Amazon S3 + Amazon CloudFront: A Match Made in the Cloud
- Utilizzo del controllo delle versioni
- Blocco degli oggetti mediante Object Lock di Amazon S
- Sharing an Amazon EBS Snapshot
- AMI condivise
- Ospitare un'applicazione a pagina singola su Amazon S3
- AWS Global Condition Keys
- Building a Data Perimeter on AWS

Video correlati:

- Securing Your Block Storage on AWS

SEC 9. In che modo proteggi i dati in transito?

Proteggi i dati in transito implementando più controlli per ridurre il rischio di accessi non autorizzati o perdita.

Best practice

- SEC09-BP01 Implementazione della gestione sicura delle chiavi e dei certificati
- SEC09-BP02 Applicazione della crittografia dei dati in transito
- SEC09-BP03 Autenticazione delle comunicazioni di rete

SEC09-BP01 Implementazione della gestione sicura delle chiavi e dei certificati

I certificati Transport Layer Security (TLS) vengono utilizzati per proteggere le comunicazioni di rete e stabilire l'identità di siti Web, risorse e carichi di lavoro su Internet, nonché sulle reti private.

Risultato desiderato: un sistema di gestione dei certificati sicuro in grado di fornire, implementare, archiviare e rinnovare i certificati in un'infrastruttura a chiave pubblica (PKI). Un meccanismo sicuro di gestione delle chiavi e dei certificati impedisce la divulgazione del materiale relativo alle chiavi private dei certificati e rinnova in automatico il certificato su base periodica. Si integra inoltre con altri servizi per fornire comunicazioni di rete e identità sicure per le risorse delle macchine all'interno del carico di lavoro. Il materiale relativo alla chiave non dovrebbe mai essere accessibile alle identità umane.

Anti-pattern comuni:

- Esecuzione di passaggi manuali durante i processi di distribuzione, implementazione o rinnovo dei certificati.
- Attenzione insufficiente alla gerarchia delle autorità di certificazione (CA) durante la progettazione di una CA privata.
- Utilizzo di certificati autofirmati per risorse pubbliche.

Vantaggi dell'adozione di questa best practice:

- Semplificazione della gestione dei certificati attraverso la distribuzione, l'implementazione e il rinnovo automatizzati
- Incoraggiamento dell'utilizzo della crittografia dei dati in transito con l'utilizzo di certificati TLS
- Maggiore sicurezza e verificabilità delle operazioni di certificazione intraprese dall'autorità di certificazione
- Organizzazione delle mansioni di gestione ai diversi livelli della gerarchia della CA

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I carichi di lavoro moderni fanno ampio uso di comunicazioni di rete crittografate utilizzando protocolli PKI come TLS. La gestione dei certificati PKI può essere complessa, ma la fornitura, la distribuzione, l'implementazione e il rinnovo automatizzati dei certificati possono ridurre gli ostacoli associati alla loro gestione.

AWS fornisce due servizi per la gestione dei certificati PKI generici: AWS Certificate Manager e AWS Autorità di certificazione privata (AWS Private CA). ACM è il servizio principale utilizzato dai clienti per fornire, gestire e implementare certificati da utilizzare in carichi di lavoro pubblici e privati AWS. ACM rilascia certificati privati mediante AWS Private CA e si integra con diversi altri servizi AWS gestiti per mettere a disposizione certificati TLS sicuri per i carichi di lavoro. ACM può rilasciare anche certificati pubblicamente attendibili da Amazon Trust Services. I certificati pubblici rilasciati da ACM possono essere utilizzati per i carichi di lavoro pubblici, poiché i browser e i sistemi operativi moderni considerano tali certificati attendibili per impostazione predefinita.

AWS Private CA consente di stabilire la propria autorità di certificazione principale o subordinata e di emettere certificati TLS tramite un'API. È possibile utilizzare questo tipo di certificati in scenari in cui si mantengono il controllo e la gestione della catena di attendibilità sul lato client della connessione TLS. Oltre ai casi d'uso TLS, AWS Private CA consente di emettere certificati per i pod Kubernetes, gli attestati dei prodotti dei dispositivi Matter, la firma del codice e altri casi d'uso che prevedono un modello personalizzato. Puoi anche usare IAM Roles Anywhere per fornire credenziali IAM temporanee ai carichi di lavoro on-premises ai quali sono stati assegnati certificati X.509 firmati dalla tua CA privata.

Oltre a ACM e AWS Private CA, AWS IoT Core fornisce supporto specializzato per il provisioning, la gestione e l'implementazione di certificati PKI su dispositivi IoT. AWS IoT Core offre meccanismi specializzati per l'onboarding dei dispositivi IoT nella tua infrastruttura chiave pubblica su larga scala.

Alcuni servizi AWS, come Gateway Amazon API ed Elastic Load Balancing, offrono funzionalità proprie per l'utilizzo dei certificati per proteggere le connessioni delle applicazioni. Ad esempio, sia Gateway API che Application Load Balancer (ALB) supportano il protocollo mTLS utilizzando certificati client creati ed esportati utilizzando la Console di gestione AWS, la CLI o le API.

Considerazioni sulla creazione di una gerarchia CA privata

Quando occorre stabilire una CA privata, è importante prestare particolare attenzione a progettare in modo corretto la gerarchia della CA fin dall'inizio. Nella creazione di una gerarchia CA privata, è consigliabile distribuire ciascun livello della gerarchia CA su Account AWS separati. Questo

passaggio intenzionale riduce l'estensione di ogni livello della gerarchia della CA, semplificando l'individuazione delle anomalie nei dati di log di CloudTrail e riducendo l'ambito di accesso o l'impatto in caso di accesso non autorizzato a uno degli account. La CA principale deve risiedere in un account separato e va utilizzata solo per l'emissione di uno o più certificati CA intermedi.

Quindi, crea una o più CA intermedie in account separati dall'account della CA principale per emettere certificati per utenti finali, dispositivi o altri carichi di lavoro. Infine, emetti certificati della tua CA principale a uso delle CA intermedie, che a loro volta emetteranno certificati per gli utenti finali o i dispositivi. Per ulteriori informazioni sulla pianificazione dell'implementazione della CA e sulla progettazione della gerarchia delle CA, inclusa la pianificazione della resilienza, la replica tra regioni, la condivisione delle CA all'interno dell'organizzazione e altro ancora, consulta Planning your AWS Private CA deployment.

Passaggi dell'implementazione

1. Determina i servizi AWS pertinenti richiesti per il tuo caso d'uso:
    - Molti casi d'uso possono sfruttare l'infrastruttura a chiave pubblica AWS esistente utilizzando AWS Certificate Manager. ACM consente di implementare certificati TLS per server Web, bilanciatori del carico o altri usi per certificati pubblicamente affidabili.
    - Prendi in considerazione AWS Private CA se occorre stabilire una gerarchia di autorità di certificazione privata o accedere a certificati esportabili. ACM può quindi essere utilizzato per emettere molti tipi di certificati di entità finale utilizzando AWS Private CA.
    - Per i casi d'uso in cui i certificati devono essere forniti su larga scala per dispositivi Internet delle cose (IoT) integrati, prendi in considerazione l'uso di AWS IoT Core.
    - Valuta la possibilità di utilizzare la funzionalità mTLS nativa in servizi come Gateway Amazon API o Application Load Balancer.
2. Implementa il rinnovo automatico dei certificati quando possibile:
    - Usa il rinnovo gestito da ACM per i certificati emessi da ACM insieme ai servizi AWS gestiti integrati.
3. Stabilisci la creazione di log e audit trail:
    - Abilita i log CloudTrail per tenere traccia degli accessi agli account che detengono le autorità di certificazione. Prendi in considerazione la possibilità di configurare la convalida dell'integrità dei file di log in CloudTrail per verificarne l'autenticità dei dati.
    - Crea e rivedi a cadenza periodica report di audit che elencano i certificati emessi o revocati dalla tua CA privata. Questi report possono essere esportati in un bucket S3.

- Quando si implementa una CA privata, è inoltre necessario creare un bucket S3 per archiviare l'elenco di revoche dei certificati (CRL). Per indicazioni sulla configurazione di questo bucket S3 in base ai requisiti del carico di lavoro, consulta Planning a certificate revocation list (CRL).

Risorse

Best practice correlate:

- SEC02-BP02 Utilizzo di credenziali temporanee
- SEC08-BP01 Implementazione della gestione sicura delle chiavi
- SEC09-BP03 Autenticazione delle comunicazioni di rete

Documenti correlati:

- How to host and manage an entire private certificate infrastructure in AWS
- How to secure an enterprise scale ACM Private CA hierarchy for automotive and manufacturing
- Private CA best practices
- How to use AWS RAM to share your ACM Private CA cross-account

Video correlati:

- Activating AWS Certificate Manager Private CA (workshop)

Esempi correlati:

- Private CA workshop
- IOT Device Management Workshop (compreso il provisioning dei dispositivi)

Strumenti correlati:

- Plugin to Kubernetes cert-manager to use AWS Private CA

SEC09-BP02 Applicazione della crittografia dei dati in transito

Applica i requisiti di crittografia definiti in base alle policy, agli obblighi normativi e agli standard dell'organizzazione per contribuire a soddisfare i requisiti organizzativi, legali e di conformità. Utilizza

solo protocolli con crittografia quando trasmetti dati sensibili al di fuori del tuo cloud privato virtuale (VPC). La crittografia aiuta a mantenere la riservatezza dei dati anche quando questi transitano su reti non affidabili.

Risultato desiderato: il traffico di rete tra le tue risorse e Internet viene crittografato per evitare l'accesso non autorizzato ai dati. Il traffico di rete nell'ambiente AWS interno viene crittografato in base ai tuoi requisiti di sicurezza. I dati in transito vengono crittografati mediante protocolli TLS sicuri e suite di crittografia.

Anti-pattern comuni:

- Utilizzo di versioni obsolete di SSL, TLS e componenti della suite di crittografia (ad esempio, SSL v3.0, chiavi RSA a 1024 bit e crittografia RC4).
- Autorizzazione del traffico non criptato (HTTP) verso o da risorse pubbliche.
- Monitoraggio e sostituzione mancati dei certificati X.509 prima della scadenza.
- Utilizzo di certificati X.509 autofirmati per TLS.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I servizi AWS forniscono endpoint HTTPS utilizzando TLS per le comunicazioni e offrono la crittografia in transito durante la comunicazione con le API AWS. I protocolli HTTP non sicuri possono essere sottoposti ad audit e bloccati in un cloud privato virtuale (VPC) tramite l'uso di gruppi di sicurezza. È possibile inoltre reindirizzare automaticamente in HTTPS le richieste HTTP in Amazon CloudFront o in un Application Load Balancer. Puoi utilizzare una policy dei bucket di Amazon Simple Storage Service (Amazon S3) per limitare la possibilità di caricare oggetti tramite HTTP, applicando efficacemente l'uso di HTTPS per il caricamento di oggetti nei tuoi bucket. Hai il controllo completo sulle tue risorse informatiche per implementare la crittografia in transito nei tuoi servizi. Inoltre, puoi utilizzare la connettività VPN nel VPC da una rete esterna o AWS Direct Connect per semplificare la crittografia del traffico. Verifica che i tuoi client stiano effettuando chiamate alle API AWS utilizzando almeno TLS 1.2, poiché l'uso di versioni di TLS precedenti a febbraio 2024 è diventato obsoleto per AWS. Consigliamo di utilizzare TLS 1.3. Se hai requisiti speciali per la crittografia dei dati in transito, puoi trovare soluzioni di terze parti nel Marketplace AWS.

Passaggi dell'implementazione

- Applica la crittografia in transito: i requisiti di crittografia definiti dovrebbero essere basati sugli standard e sulle best practice più recenti e consentire solo protocolli sicuri. Ad esempio, configura

un gruppo di sicurezza per consentire solo il protocollo HTTPS a un Application Load Balancer o a un'istanza Amazon EC2.

- Configura protocolli sicuri nei servizi edge: configura HTTPS con Amazon CloudFront e utilizza un profilo di sicurezza adeguato al tuo livello di sicurezza e il tuo caso d'uso.
- Usa una VPN per la connettività esterna: valuta l'impiego di una VPN IPsec per la protezione delle connessioni punto a punto o rete a rete al fine di garantire la riservatezza e l'integrità dei dati.
- Configura protocolli sicuri nei bilanciatori del carico: seleziona una policy di sicurezza che fornisca le suite di crittografia più solide supportate dai client che si connetteranno al listener. Create an HTTPS listener for your Application Load Balancer.
- Configura protocolli di sicurezza in Amazon Redshift: configura il cluster per richiedere una connessione Secure Socket Layer (SSL) o Transport Layer Security (TLS)..
- Configura protocolli sicuri: consulta la documentazione del servizio AWS per determinare le funzionalità di crittografia in transito.
- Configura l'accesso sicuro durante il caricamento su bucket Amazon S3: utilizza i controlli delle policy sui bucket Amazon S3 per applicare l'accesso sicuro ai dati.
- Prendi in considerazione l'utilizzo di AWS Certificate Manager: ACM ti consente di fornire, gestire e implementare certificati TLS pubblici da utilizzare con i servizi AWS.
- Prendi in considerazione l'utilizzo AWS Autorità di certificazione privata per le esigenze di PKI private: AWS Private CA consente di creare gerarchie di autorità di certificazione (CA) private per emettere certificati X.509 di entità finale, utilizzabili per creare canali TLS crittografati.

Risorse

Documenti correlati:

- Using HTTPS with CloudFront
- Connect your VPC to remote networks using AWS Virtual Private Network
- Create an HTTPS listener for your Application Load Balancer
- Tutorial: Configure SSL/TLS on Amazon Linux 2
- Using SSL/TLS to encrypt a connection to a DB instance
- Configuring security options for connections

SEC09-BP03 Autenticazione delle comunicazioni di rete

Verifica l'identità delle comunicazioni utilizzando protocolli che supportano l'autenticazione, ad esempio Transport Layer Security (TLS) o IPsec.

Progetta il carico di lavoro in modo da utilizzare protocolli di rete sicuri e autenticati per le comunicazioni tra servizi, applicazioni o utenti. L'utilizzo di protocolli di rete che supportano autenticazione e autorizzazione offre un controllo più rigido sui flussi di rete e riduce l'impatto di eventuali accessi non autorizzati.

Risultato desiderato: un carico di lavoro con un piano dati ben definito e flussi di traffico del piano di controllo (control-plane) tra i servizi. I flussi di traffico utilizzano protocolli di rete autenticati e crittografati laddove tecnicamente fattibile.

Anti-pattern comuni:

- Flussi di traffico non crittografati o non autenticati all'interno del carico di lavoro.
- Riutilizzo delle credenziali di autenticazione tra più utenti o entità.
- Uso esclusivo di controlli di rete come meccanismo di controllo degli accessi.
- Creazione di un meccanismo di autenticazione personalizzato anziché usare meccanismi di autenticazione standard del settore.
- Flussi di traffico eccessivamente permissivi tra i componenti del servizio o altre risorse nel VPC.

Vantaggi dell'adozione di questa best practice:

- Limita l'ambito dell'impatto di eventuali accessi non autorizzati a una parte del carico di lavoro.
- Fornisce un livello maggiore di sicurezza affinché solo entità autenticate eseguano le azioni.
- Migliora il disaccoppiamento dei servizi definendo e applicando in modo chiaro le interfacce di trasferimento dei dati previste.
- Migliora monitoraggio, creazione di log e risposta agli incidenti tramite l'attribuzione di richieste e interfacce di comunicazione ben definite.
- Fornisce una difesa approfondita ai carichi di lavoro combinando i controlli di rete con quelli di autenticazione e autorizzazione.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

È possibile suddividere i modelli di traffico di rete del tuo carico di lavoro in due categorie:

- Il traffico est-ovest corrisponde ai flussi di traffico tra i servizi facenti parte di un carico di lavoro.
- Il traffico nord-sud rappresenta i flussi di traffico tra carico di lavoro e consumatori.

Sebbene crittografare il traffico nord-sud sia la prassi comune, proteggere il traffico est-ovest mediante protocolli autenticati non è così frequente. Le moderne best practice di sicurezza raccomandano che la progettazione della rete non sia l'unico elemento in grado di garantire una relazione affidabile tra due entità. Quando due servizi possono trovarsi all'interno di una rete comune, è comunque consigliabile crittografare, autenticare e autorizzare le comunicazioni tra tali servizi.

Ad esempio, le API del servizio AWS utilizzano il protocollo di firma AWSSignature Version 4 (SIGv4) per autenticare il chiamante, indipendentemente dalla rete di provenienza della richiesta. Questa autenticazione garantisce che le API di AWS possano verificare l'identità che ha richiesto l'azione e che tale identità possa quindi essere combinata con le policy per decidere se autorizzare o meno l'azione.

Servizi come Amazon VPC Lattice e Gateway Amazon API consentono di utilizzare lo stesso protocollo di firma SigV4 per aggiungere autenticazione e autorizzazione al traffico est-ovest nei propri carichi di lavoro. Se le risorse esterne al tuo ambiente AWS devono comunicare con servizi che richiedono autenticazione e autorizzazione basate su SigV4, puoi utilizzare AWS Identity and Access Management (IAM) Roles Anywhere sulla risorsa non AWS per acquisire credenziali AWS temporanee. Queste credenziali possono essere utilizzate per firmare richieste ai servizi che utilizzano SigV4 per autorizzare l'accesso.

Un altro meccanismo comune per l'autenticazione del traffico est-ovest è l'autenticazione reciproca TLS (mTLS). Molte applicazioni Internet delle cose (IoT), business-to-business (B2B) e microservizi utilizzano mTLS per convalidare l'identità di entrambi i lati di una comunicazione TLS mediante l'uso di certificati X.509 lato client e lato server. Questi certificati possono essere emessi da AWS Autorità di certificazione privata (AWS Private CA). Puoi utilizzare servizi come Gateway Amazon API per garantire l'autenticazione mTLS per le comunicazioni interne ai carichi di lavoro o tra un carico di lavoro e un altro. Application Load Balancer supporta mTLS anche per i carichi di lavoro interni o esterni. Sebbene fornisca informazioni di autenticazione per entrambi i lati di una comunicazione TLS, mTLS non fornisce un meccanismo di autorizzazione.

Infine, OAuth 2.0 e OpenID Connect (OIDC) sono due protocolli in genere utilizzati per controllare l'accesso ai servizi da parte degli utenti, ma stanno diventando sempre più diffusi anche per il traffico

da servizio a servizio. API Gateway fornisce un sistema di autorizzazione JSON Web token (JWT), che consente ai carichi di lavoro di limitare l'accesso ai percorsi API utilizzando JWT emessi da gestori dell'identità digitale OIDC o OAuth 2.0. È possibile utilizzare gli ambiti OAuth2 come base per decisioni di autorizzazione essenziali, ma i controlli di autorizzazione vanno comunque implementati a livello di applicazione. Gli ambiti OAuth2 da soli non possono supportare requisiti di autorizzazione più complessi.

Passaggi dell'implementazione

- Definisci e documenta i flussi di rete del carico di lavoro: il primo passo per implementare una strategia di difesa approfondita consiste nel definire i flussi di traffico del carico di lavoro.
  - Crea un diagramma del flusso di dati che definisca in modo chiaro le modalità di trasmissione dei dati tra i diversi servizi che costituiscono il carico di lavoro. Questo diagramma è il primo passo per autorizzare tali flussi nei canali di rete autenticati.
  - Nelle fasi di sviluppo e test dota il carico di lavoro di strumenti per controllare che il diagramma del flusso di dati rifletta in modo preciso il comportamento del carico di lavoro in fase di runtime.
  - Un diagramma di flusso di dati può essere utile anche quando si esegue un esercizio di modellazione delle minacce, come illustrato in SEC01-BP07 Identificare le minacce e dare priorità alle mitigazioni utilizzando un modello di minaccia.
- Stabilisci i controlli di rete: valuta le funzionalità di AWS per stabilire controlli di rete allineati ai flussi di dati. Sebbene non debbano costituire l'unico elemento di controllo della sicurezza, i confini della rete forniscono un livello nella strategia di difesa di alto profilo a protezione del carico di lavoro.
  - Utilizza i gruppi di sicurezza per stabilire, definire e limitare i flussi di dati tra le risorse.
  - Valuta l'utilizzo di AWS PrivateLink per comunicare sia con servizi AWS e di terze parti che supportano AWS PrivateLink. I dati inviati tramite un endpoint di interfaccia AWS PrivateLink rimangono all'interno della dorsale della rete AWS e non attraversano la rete Internet pubblica.
- Implementa autenticazione e autorizzazione tra i servizi del tuo carico di lavoro: scegli il set di servizi AWS più adeguato a fornire flussi di traffico autenticati e crittografati nel tuo carico di lavoro.
  - Prendi in considerazione Amazon VPC Lattice per proteggere la comunicazione da servizio a servizio. VPC Lattice può utilizzare l'autenticazione SigV4 in combinazione con le policy di autenticazione per controllare l'accesso da servizio a servizio.
  - Per la comunicazione da servizio a servizio tramite mTLS, prendi in considerazione API Gateway o Application Load Balancer. AWS Private CA consente di stabilire una gerarchia CA privata in grado di emettere certificati da utilizzare con mTLS.

  - In caso di integrazione con servizi che utilizzano OAuth 2.0 o OIDC, prendi in considerazione API Gateway mediante il sistema di autorizzazione JWT.
  - Per la comunicazione tra il carico di lavoro e i dispositivi IoT, prendi in considerazione AWS IoT Core, che offre diverse opzioni per la crittografia e l'autenticazione del traffico di rete.
- Monitora gli accessi non autorizzati: monitora in modo continuo i canali di comunicazione non intenzionali, i tentativi di accesso dei principali non autorizzati a risorse protette e altri schemi di accesso impropri.
  - Se utilizzi VPC Lattice per la gestione dell'accesso ai tuoi servizi, prendi in considerazione l'abilitazione e il monitoraggio dei log di accesso VPC Lattice. Questi log di accesso includono informazioni sull'entità richiedente, informazioni di rete tra cui VPC di origine e destinazione e metadati della richiesta.
  - Considera l'abilitazione dei log di flusso VPC per acquisire metadati sui flussi di rete e verificare a cadenza periodica la presenza di anomalie.
  - Consulta la AWS Security Incident Response Guide e la sezione Risposta agli imprevisti del pilastro della sicurezza del Framework AWS Well-Architected per ulteriori indicazioni su pianificazione, simulazione e risposta agli incidenti di sicurezza.

Risorse

Best practice correlate:

- SEC03-BP07 Analisi dell'accesso multi-account e pubblico
- SEC02-BP02 Utilizzo di credenziali temporanee
- SEC01-BP07 Identificare le minacce e dare priorità alle mitigazioni utilizzando un modello di minaccia

Documenti correlati:

- Evaluating access control methods to secure Amazon API Gateway APIs
- Configuring mutual TLS authentication for a REST API
- How to secure API Gateway HTTP endpoints with JWT authorizer
- Authorizing direct calls to AWS services using AWS IoT Core credential provider
- AWS Security Incident Response Guide

Video correlati:

- AWS re:invent 2022: Introducing VPC Lattice
- AWS re:invent 2020: Serverless API authentication for HTTP APIs on AWS

Esempi correlati:

- Amazon VPC Lattice Workshop
- Zero-Trust Episode 1 – The Phantom Service Perimeter workshop

Risposta agli incidenti

Domanda

- SEC 10. In che modo è possibile prevedere gli incidenti, rispondere agli stessi e risolverli?

SEC 10. In che modo è possibile prevedere gli incidenti, rispondere agli stessi e risolverli?

Anche se dispone di controlli preventivi e di rilevamento maturi, l'organizzazione deve ancora implementare meccanismi per rispondere e mitigare il potenziale impatto degli incidenti di sicurezza. La tua preparazione influisce fortemente sulla capacità dei team di operare in modo efficace durante un incidente, isolare, contenere ed eseguire indagini sui problemi e ripristinare le operazioni a uno stato valido noto. La messa in atto degli strumenti e l'accesso prima di un incidente di sicurezza, quindi la pratica sistematica della risposta agli incidenti durante le giornate di gioco, aiuterà a garantire il ripristino, riducendo al minimo le interruzioni dell'attività.

Best practice

- SEC10-BP01 Identificazione del personale chiave e delle risorse esterne
- SEC10-BP02 Sviluppo di piani di gestione degli incidenti
- SEC10-BP03 Preparazione di funzionalità forensi
- SEC10-BP04 Sviluppo e test di playbook di risposta agli incidenti di sicurezza
- SEC10-BP05 Preassegnazione dell'accesso
- SEC10-BP06 Implementazione anticipata degli strumenti
- SEC10-BP07 Esecuzione di simulazioni

- SEC10-BP08 Definizione di un framework per apprendere dagli incidenti

SEC10-BP01 Identificazione del personale chiave e delle risorse esterne

Identifica personale, risorse e requisiti legali interni ed esterni per consentire all'organizzazione a rispondere a un incidente.

Risultato desiderato: presenza di un elenco del personale chiave, delle relative informazioni di contatto e dei ruoli svolti nel rispondere a un evento di sicurezza. Rivedi queste informazioni con regolarità e aggiornarle per riflettere i cambiamenti del personale dal punto di vista degli strumenti interni ed esterni. Nel documentare queste informazioni, prendi in considerazione tutti i fornitori di servizi e i venditori di terze parti, compresi partner di sicurezza, fornitori di cloud e applicazioni software-as-a-service (SaaS). Durante un evento di sicurezza, il personale è disponibile con il livello di responsabilità, il contesto e l'accesso appropriati per poter rispondere ed eseguire il ripristino.

Anti-pattern comuni:

- Mancata tenuta di un elenco aggiornato del personale chiave con le informazioni di contatto, i ruoli e le responsabilità in caso di risposta a eventi di sicurezza.
- Si presume che tutti conoscano persone, dipendenze, infrastruttura e soluzioni per rispondere a un evento ed eseguire il ripristino dopo lo stesso.
- Mancata predisposizione di un archivio di documenti o conoscenze che rappresenti l'infrastruttura o la progettazione di applicazioni chiave.
- Mancata predisposizione di processi di onboarding adeguati per i nuovi dipendenti, in modo che possano contribuire in modo efficace alla risposta a un evento di sicurezza, come la realizzazione di simulazioni di eventi.
- Mancata predisposizione di un percorso di escalation quando il personale chiave è temporaneamente non disponibile o non risponde durante gli eventi di sicurezza.

Vantaggi dell'adozione di questa best practice: riduzione del tempo di valutazione e risposta impiegato per identificare il personale giusto e il relativo ruolo durante un evento grazie a questa pratica. Riduci al minimo le perdite di tempo durante un evento mantenendo un elenco aggiornato del personale chiave e dei relativi ruoli, in modo da poter portare le persone giuste al triage e al ripristino da un evento.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Identifica il personale chiave all'interno dell'organizzazione: conserva un elenco di contatti del personale interno alla tua organizzazione che potrebbe essere necessario coinvolgere. Rivedi e aggiorna in modo regolare queste informazioni in caso di spostamento del personale, quali modifiche organizzative, promozioni e cambi di team. Questo è particolarmente importante per i ruoli chiave come gli incident manager, i team di risposta e i responsabili delle comunicazioni.

- Responsabile degli incidenti: i responsabili degli incidenti dispongono dell'autorità generale durante la risposta all'evento.
- Persone che intervengono dopo un incidente: le persone che intervengono dopo un incidente sono responsabili delle attività di indagine e correzione. Queste persone possono differire in base al tipo di evento, ma in genere sono sviluppatori e team operativi responsabili dell'applicazione interessata.
- Responsabile delle comunicazioni: il responsabile delle comunicazioni gestisce comunicazioni interne ed esterne, in particolare con gli enti pubblici, le autorità di regolamentazione e i clienti.
- Processo di onboarding: attività periodiche di formazione e onboarding per i nuovi dipendenti, mirate a fornire le competenze e le conoscenze necessarie per dare un contributo efficace alle iniziative di risposta agli incidenti. Include simulazioni ed esercizi pratici nell'ambito del processo di onboarding per facilitarne la preparazione.
- Esperti in materia (SME): in caso di team distribuiti e autonomi, ti consigliamo di identificare un SME per carichi di lavoro mission critical. Queste persone offrono approfondimenti su funzionamento e classificazione dei dati dei carichi di lavoro critici coinvolti nell'evento.

Formato di tabella di esempio:

```
  | Role | Name | Contact Information | Responsibilities |
1 | ——- | ——- | ——- | ——- |
2 | Incident Manager | Jane Doe| jane.doe@example.com | Overall authority during
 response |
3 | Incident Responder | John Smith | john.smith@example.com | Investigation and
 remediation |
4 | Communications Lead | Emily Johnson | emily.johnson@example.com | Internal and
 external communications |
5 | Communications Lead | Michael Brown | michael.brown@example.com | Insights on
 critical workloads |
```

Prendi in considerazione l'utilizzo della funzionalità AWS Systems Manager Incident Manager per l'acquisizione dei contatti chiave, la definizione di un piano di risposta, l'automazione degli orari delle chiamate e la creazione di piani di escalation. Automatizza e organizza i turni per tutto il personale attraverso un programma di chiamata, in modo che la responsabilità del carico di lavoro sia condivisa tra i proprietari. Ciò promuove buone pratiche, come l'emissione di metriche e log pertinenti e la definizione di soglie di allarme importanti per il carico di lavoro.

Identifica i partner esterni: le aziende utilizzano strumenti creati da fornitori di software indipendenti (ISV), partner e subappaltatori per creare soluzioni differenziate per i propri clienti. Coinvolgi il personale chiave di queste parti che può aiutarti a rispondere e a eseguire il ripristino dopo un incidente. Ti consigliamo di iscriverti al livello appropriato di Supporto per ottenere un rapido accesso agli SME AWS attraverso un caso di supporto. Prendi in considerazione accordi simili con tutti i fornitori di soluzioni critiche per i carichi di lavoro. Alcuni eventi di sicurezza richiedono alle aziende quotate in borsa di notificare evento ed effetti agli enti pubblici e alle autorità di regolamentazione pertinenti. Mantieni e aggiorna le informazioni di contatto per i dipartimenti pertinenti e le persone responsabili.

Passaggi dell'implementazione

1. Configura una soluzione per la gestione degli incidenti.
    a. Prendi in considerazione l'implementazione di Incident Manager nel tuo account Security Tooling.
2. Definisci i contatti nella tua soluzione di gestione degli incidenti.
    a. Definisci almeno due tipi di canali per ogni contatto (come SMS, telefono o e-mail), per garantire la raggiungibilità durante un incidente.
3. Definisci un piano di risposta.
    a. Identifica i contatti più opportuni da coinvolgere durante un incidente. Definisci piani di escalation in linea con i ruoli del personale da coinvolgere, piuttosto che con i singoli contatti. Valuta la possibilità di includere i contatti che potrebbero essere responsabili dell'informare entità esterne, anche se non direttamente coinvolti nella risoluzione dell'incidente.

Risorse

Best practice correlate:

- OPS02-BP03 Assegnazione di proprietari identificati alle operazioni che siano responsabili delle relative prestazioni

Documenti correlati:

- AWS Security Incident Response Guide

Esempi correlati:

- AWS Framework per playbook per i clienti
- Prepare for and respond to security incidents in your AWS environment

Strumenti correlati:

- AWS Systems Manager Incident Manager

Video correlati:

- Amazon's approach to security during development

SEC10-BP02 Sviluppo di piani di gestione degli incidenti

Il primo documento da predisporre per la risposta agli incidenti è il piano di risposta agli incidenti. Lo scopo del piano di risposta agli incidenti è costituire la base del programma e della strategia di risposta agli incidenti.

Vantaggi dell'adozione di questa best practice: lo sviluppo di processi di risposta agli incidenti completi e definiti in modo chiaro è fondamentale per un programma di risposta agli incidenti efficace e scalabile. Quando si verifica un evento di sicurezza, passaggi e flussi di lavoro ben definiti agevoleranno una risposta tempestiva. Potrebbero essere già presenti processi di risposta agli incidenti. Indipendentemente dallo stato attuale, è importante aggiornare, iterare e testare con regolarità i processi di risposta agli incidenti.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Un piano di gestione degli incidenti è fondamentale per rispondere, mitigare ed eseguire il ripristino a seguito del potenziale impatto degli incidenti di sicurezza. Un piano di gestione degli incidenti è un processo strutturato volto a identificare, correggere e rispondere tempestivamente agli incidenti di sicurezza.

Il cloud presenta molti degli stessi ruoli e requisiti operativi che si trovano in un ambiente on-premises. Nella creazione di un piano di gestione degli incidenti, è importante tenere conto delle strategie di risposta e ripristino ideali per i risultati aziendali e ai requisiti di conformità. Ad esempio, se gestisci carichi di lavoro in AWS conformi a FedRAMP negli Stati Uniti, occorre seguire le raccomandazioni enunciate nel documento NIST SP 800-61 Computer Security Handling Guide. Allo stesso modo, quando gestisci carichi di lavoro che memorizzano informazioni di identificazione personale (PII), valuta come proteggere e rispondere ai problemi relativi alla residenza e all'utilizzo dei dati.

Quando crei un piano di gestione degli incidenti per i tuoi carichi di lavoro in AWS, inizia con il modello di responsabilità condivisa AWS per creare un approccio di difesa approfondito alla risposta agli incidenti. In questo modello, AWS gestisce la sicurezza del cloud e tu sei responsabile della sicurezza nel cloud. Ciò significa che mantieni il controllo e sei responsabile dei controlli di sicurezza che scegli di implementare. La AWS Security Incident Response Guide illustra concetti chiave e linee guida di base per la creazione di un piano di gestione degli incidenti incentrato sul cloud.

Un piano di gestione degli incidenti efficace va iterato in modo continuo per rimanere in linea con l'obiettivo delle operazioni cloud. Prendi in considerazione l'utilizzo dei piani di implementazione illustrati di seguito durante la creazione e l'evoluzione del tuo piano di gestione degli incidenti.

Passaggi dell'implementazione

1. Definisci ruoli e responsabilità all'interno dell'organizzazione per la gestione degli eventi di sicurezza. Il processo dovrebbe coinvolgere rappresentanti di vari dipartimenti, tra cui:
   - Risorse umane (HR)
   - Team esecutivo
   - Ufficio legale
   - Proprietari e sviluppatori di applicazioni (SME, ossia esperti in materia)
2. Determina in modo chiaro i soggetti RACI (Responsible, Accountable, Consulted, and Informed) da tenere in considerazione in caso di incidente. Crea un grafico RACI per facilitare una comunicazione rapida e diretta, e delinea chiaramente la leadership nelle diverse fasi di un evento.
3. Coinvolgi i proprietari e gli sviluppatori delle applicazioni (SME) durante un incidente, poiché tali soggetti possono fornire informazioni e contesto preziosi per aiutare a misurare l'impatto. Instaura relazioni con questi SME e fai pratica con loro utilizzando vari scenari di risposta agli incidenti prima che si verifichi un incidente reale.
4. Coinvolgi partner attendibili o esperti esterni nel processo di indagine o risposta, poiché tali soggetti possono fornire competenze e prospettive aggiuntive.

5. Allinea i piani e i ruoli di gestione degli incidenti alle normative locali o ai requisiti di conformità che regolano la tua organizzazione.
6. Effettua regolarmente esercitazioni pratiche e test sui piani di risposta agli incidenti, coinvolgendo tutti i ruoli e le responsabilità definiti. Questo aiuta a semplificare il processo e ad assicurarsi di avere una risposta coordinata ed efficiente agli incidenti di sicurezza.
7. Rivedi e aggiorna i ruoli, le responsabilità e il grafico RACI periodicamente o man mano che la struttura organizzativa o i requisiti cambiano.

Analizza il supporto e i team di risposta di AWS

- Supporto AWS
  - Supporto offre un'ampia gamma di piani che forniscono accesso agli strumenti e alla competenza che genera successo e stato operativo delle soluzioni AWS. Se ti occorre supporto tecnico e ulteriori risorse per pianificare, implementare e ottimizzare il tuo ambiente AWS, puoi selezionare il piano di supporto più adatto al tuo caso d'uso AWS.
  - Considera il Centro supporto in Console di gestione AWS (è richiesto l'accesso) come punto di contatto centralizzato per assistenza circa problemi relativi alle tue risorse AWS. L'accesso a Supporto è controllato da AWS Identity and Access Management. Per ulteriori informazioni sull'accesso alle funzionalità Supporto, consulta Getting started with Supporto.
- AWS Team di risposta agli incidenti dei clienti (CIRT)
  - Il Team di risposta agli incidenti dei clienti AWS (CIRT) è un team AWS globale specializzato, disponibile 24 ore su 24, 7 giorni su 7, che fornisce supporto ai clienti durante eventi di sicurezza attivi sul lato cliente del modello di responsabilità condivisa di AWS.
  - Quando il team AWS CIRT ti supporta, fornisce assistenza nella valutazione e nel ripristino di un evento di sicurezza su AWS. Può fornire assistenza nell'analisi delle cause principali grazie all'uso dei log dei servizi AWS e fornire suggerimenti per il ripristino. Può altresì fornire consigli e best practice sulla sicurezza così da evitare eventi di sicurezza in futuro.
  - I clienti AWS possono rivolgersi al team AWS CIRT attraverso un caso Supporto.
- AWS Security Incident Response
  - Annunciato al re:Invent 2024, AWS Security Incident Response è un servizio gestito di risposta agli incidenti di sicurezza che utilizza sia la moderna tecnologia di triage che un operatore umano presente nel loop. Il servizio acquisisce tutti gli esiti di GuardDuty e tutti gli esiti di terze parti inviati a AWS Security Hub CSPM per la valutazione al fine di avvisare il cliente solo degli esiti che richiedono un'indagine. Il servizio fornisce anche un portale per presentare casi reattivi

in caso di un evento di sicurezza notato dal cliente e ricevere supporto dal team di risposta avanzata agli incidenti di AWS.

- Supporto per la risposta agli attacchi DDoS
  - AWS offre AWS Shield, un servizio gestito di protezione da attacchi di tipo DDoS (Distributed Denial of Service) che protegge le applicazioni Web in esecuzione in AWS. Shield fornisce un rilevamento continuo e prevenzione incorporata automatica che riducono al minimo il tempo di inattività e la latenza dell'applicazione, così da non dover rivolgersi al Supporto per beneficiare della protezione DDoS. I livelli esistenti di Shield sono due: AWS Shield Standard e AWS Shield Advanced. Per maggiori informazioni sulle differenze tra questi due livelli, consulta la documentazione della funzionalità Shield.
- AWS Managed Services (AMS)
  - AWS Managed Services (AMS) offre una gestione continua dell'infrastruttura AWS, così potrai concentrarti solo sulle tue applicazioni. Grazie all'implementazione di best practice per la manutenzione dell'infrastruttura, AMS consente di ridurre rischi e costi operativi. AMS automatizza attività frequenti quali richieste di modifica, monitoraggio, gestione di patch, sicurezza e backup, nonché fornisce servizi completi per il ciclo di vita per gestire provisioning, esecuzione e supporto dell'infrastruttura.
  - AMS è responsabile dell'implementazione di una suite di controlli di sicurezza e fornisce una risposta di prima linea agli avvisi 24 ore su 24, 7 giorni su 7. In caso di avviso, AMS si attiene a una serie standard di playbook automatici e manuali per verificare una risposta coerente. Questi playbook vengono condivisi con i clienti AMS durante l'onboarding in modo che possano sviluppare e coordinare una risposta con AMS.

Sviluppo di piani di risposta agli incidenti

Lo scopo del piano di risposta agli incidenti è costituire la base del programma e della strategia di risposta agli incidenti. Il piano di risposta agli incidenti deve essere contenuto in un documento formale. Un piano di risposta agli incidenti include in genere le seguenti sezioni:

- Una panoramica del team di risposta agli incidenti: delinea obiettivi e funzioni del team di risposta agli incidenti.
- Ruoli e responsabilità: indica le parti interessate alla risposta agli incidenti e illustra in dettaglio i loro ruoli in caso di incidente.
- Un piano di comunicazione: fornisce dettagli sulle informazioni di contatto e sulle tue modalità di comunicazione durante un incidente.

- Metodi di comunicazione di backup: è consigliabile utilizzare la comunicazione fuori banda come backup in caso di incidente. Un esempio di applicazione che fornisce un canale di comunicazione fuori banda sicuro è AWS Wickr.
- Fasi di risposta agli incidenti e azioni da intraprendere: elenca le fasi della risposta agli incidenti (ad esempio, rilevamento, analisi, eliminazione, contenimento e ripristino), comprese le azioni di alto livello da intraprendere all'interno di tali fasi.
- Definizioni di gravità e assegnazione della priorità agli incidenti: illustra in dettaglio come classificare la gravità di un incidente, le modalità di assegnazione della priorità all'incidente e, quindi, in che modo le definizioni di gravità influiscono sulle procedure di escalation.

Sebbene queste sezioni siano comuni ad aziende di diverse dimensioni e settori, il piano di risposta agli incidenti di ciascuna organizzazione è unico. Devi creare un piano di risposta agli incidenti che funzioni al meglio per la tua organizzazione.

Risorse

Best practice correlate:

- SEC04 Rilevamento

Documenti correlati:

- AWS Security Incident Response Guide
- NIST: Computer Security Incident Handling Guide

SEC10-BP03 Preparazione di funzionalità forensi

Prima che si verifichi un incidente di sicurezza, puoi sviluppare funzionalità forensi per supportare le indagini sugli eventi di sicurezza.

Livello di rischio associato se questa best practice non fosse adottata: medio

Il concetto della tradizionale analisi forense on-premises si applica ad AWS. Per informazioni chiave su come iniziare a sviluppare funzionalità forensi in Cloud AWS, consulta Forensic investigation environment strategies in the Cloud AWS.

Una volta configurati ambiente e struttura di Account AWS per le funzionalità forensi, definisci le tecnologie necessarie in modo da eseguire in modo ottimale le metodologie forensi in quattro fasi:

- Raccolta: raccogli i log AWS pertinenti, come quelli di AWS CloudTrail, AWS Config, del flusso VPC e dell'host. Raccogli snapshot, backup e dump di memoria delle risorse AWS interessate, se disponibili.
- Esame: rivedi i dati raccolti estraendo e valutando le informazioni pertinenti.
- Analisi: analizza i dati raccolti per comprendere l'incidente e trarre le conclusioni.
- Creazione di report: presenta le informazioni risultanti dalla fase di analisi.

Passaggi dell'implementazione

Preparazione dell'ambiente per le funzionalità forensi

AWS Organizations ti aiuta a gestire e dirigere a livello centrale un ambiente AWS mentre le risorse AWS crescono e scalano. Un'organizzazione AWS consolida gli Account AWS in modo da poterli amministrare come una singola unità. Puoi utilizzare le unità organizzative (UO) per raggruppare gli account e amministrarli come singola unità.

Per rispondere agli incidenti, è utile disporre di una struttura di Account AWS che supporti le funzioni di risposta agli incidenti e includa un'unità organizzativa di sicurezza e un'unità organizzativa con funzionalità forensi. All'interno dell'unità organizzativa di sicurezza, è necessario disporre degli account per:

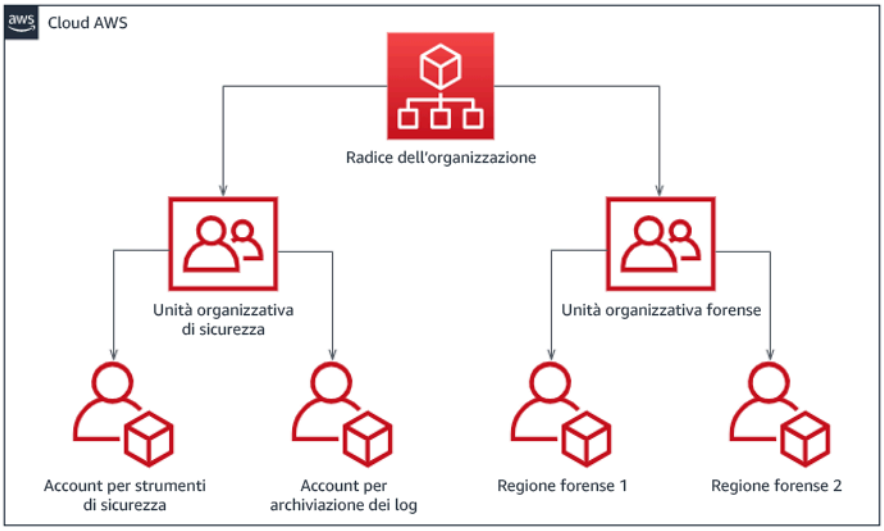
- Archiviazione dei log: aggrega i log in un Account AWS di archiviazione dei log con autorizzazioni limitate.
- Strumenti di sicurezza: centralizza i servizi di sicurezza in un Account AWS dello strumento di sicurezza. Questo account funge da amministratore delegato per i servizi di sicurezza.

Nell'unità organizzativa con funzionalità forensi, puoi implementare uno o più account con funzionalità forensi per ciascuna regione in cui operi, a seconda di quale è più adatta all'azienda e al modello operativo. Se crei un account con funzionalità forensi per regione, puoi bloccare la creazione di risorse AWS al di fuori della regione e ridurre il rischio di copia delle risorse in una regione indesiderata. Ad esempio, se operi solo nella regione degli Stati Uniti orientali (Virginia settentrionale) (`us-east-1`) e Stati Uniti occidentali (Oregon) (`us-west-2`), nell'unità organizzativa con funzionalità forensi avrai due account: uno per `us-east-1` e uno per `us-west-2`.

Puoi creare un Account AWS con funzionalità forensi per più regioni. Quando si copiano le risorse AWS nell'account, presta attenzione a rispettare i requisiti di sovranità dei dati. Poiché la creazione di nuovi account richiede tempo, è fondamentale creare e fornire gli strumenti adatti agli account con

funzionalità forensi con largo anticipo rispetto agli incidenti, in modo che gli addetti siano preparati a utilizzarli in modo efficace per la risposta.

Il diagramma seguente mostra una struttura degli account di esempio che include un'unità organizzativa con funzionalità forensi con account con funzionalità forensi per regione:



Struttura degli account per regione per la risposta agli incidenti

Acquisizione di backup e snapshot

La configurazione dei backup di sistemi e database importanti è fondamentale per il ripristino da un incidente di sicurezza e per scopi forensi. Grazie ai backup puoi ripristinare i tuoi sistemi allo stato di sicurezza precedente. In AWS puoi acquisire snapshot di varie risorse. Gli snapshot forniscono i backup point-in-time delle risorse. Esistono molti servizi AWS che offrono supporto nelle operazioni di backup e ripristino. Per informazioni dettagliate su questi servizi e approcci per il backup e il ripristino, consulta la guida prescrittiva per il backup e il ripristino e Use backups to recover from security incidents.

Soprattutto in situazioni come un attacco ransomware, è fondamentale che i backup siano ben protetti. Per indicazioni sulla protezione dei backup, consulta Top 10 security best practices for securing backups in AWS. Oltre a proteggere i backup, è necessario sottoporli regolarmente a processi di backup e ripristino per verificare che tecnologia e procedure in uso funzionino come previsto.

Automazione delle funzionalità forensi

Durante un evento di sicurezza, il team addetto a rispondere agli incidenti deve essere in grado di raccogliere e analizzare rapidamente le prove, mantenendo la precisione per il periodo di tempo

relativo all'evento (ad esempio, acquisendo i log relativi a una risorsa o un evento specifico o raccogliendo il dump della memoria di un'istanza Amazon EC2). Per il team addetto a rispondere agli incidenti è difficile e dispendioso in termini di tempo raccogliere manualmente le prove pertinenti, soprattutto se istanze e account sono numerosi. Inoltre, la raccolta manuale può essere soggetta all'errore umano. Per questi motivi, occorre sviluppare e implementare il più possibile l'automazione per le funzionalità forensi.

AWS offre una serie di risorse di automazione per le funzionalità forensi, elencate nella sezione Risorse più avanti. Queste risorse sono esempi di modelli di funzionalità forensi che abbiamo sviluppato, implementate dai clienti Sebbene costituiscano un'utile architettura di riferimento per iniziare, prendi in considerazione la possibilità di modificarli o creare nuovi modelli di automazione per le funzionalità forensi in base ad ambiente, requisiti, strumenti e processi forensi.

Risorse

Documenti correlati:

- AWS Security Incident Response Guide - Develop Forensics Capabilities
- AWS Security Incident Response Guide - Forensics Resources
- Forensic investigation environment strategies in the Cloud AWS
- How to automate forensic disk collection in AWS
- AWS Prescriptive Guidance - Automate incident response and forensics

Video correlati:

- Automating Incident Response and Forensics

Esempi correlati:

- Automated Incident Response and Forensics Framework
- Automated Forensics Orchestrator for Amazon EC2

SEC10-BP04 Sviluppo e test di playbook di risposta agli incidenti di sicurezza

Una parte fondamentale della preparazione dei processi di risposta agli incidenti è costituita dalla predisposizione di playbook. I playbook di risposta agli incidenti forniscono indicazioni prescrittive e

passaggi da seguire in caso di evento di sicurezza. Una struttura e passaggi chiari semplificano la risposta e riducono la probabilità di errore umano.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

È necessario creare i playbook per scenari di incidenti come:

- Incidenti previsti: i playbook devono essere creati per gli incidenti previsti, tra cui minacce come Denial of Service (DoS), ransomware e la compromissione delle credenziali.
- Avvisi o esiti di sicurezza noti: i playbook devono essere creati per affrontare gli esiti e gli avvisi di sicurezza noti, ad esempio quelli di Amazon GuardDuty. Quando ricevi un esito di GuardDuty, il playbook dovrebbe fornire istruzioni chiare per evitare che l'avviso venga gestito in modo errato o ignorato. Per ulteriori dettagli e indicazioni sulla riparazione, consulta Correzione dei problemi di sicurezza rilevati da GuardDuty.

I playbook devono contenere i passaggi tecnici che un analista della sicurezza deve seguire per indagare e rispondere in modo adeguato a un potenziale incidente di sicurezza.

Il Customer Incident Response Team (CIRT) di AWS ha pubblicato un repository GitHub contenente i playbook di risposta agli incidenti, organizzati per scenario, tipo e risorsa delle minacce. Questi playbook possono essere adattati per allinearsi alle procedure di risposta agli incidenti esistenti o fungere da base per svilupparne di nuove.

Passaggi dell'implementazione

Gli elementi da includere in un playbook sono:

- Panoramica del playbook: quale scenario di rischio o incidente affronta questo playbook? Qual è l'obiettivo del playbook?
- Prerequisiti: quali log, meccanismi di rilevamento e strumenti automatizzati sono necessari per questo scenario di incidente? Qual è la notifica prevista?
- Informazioni su comunicazione ed escalation: chi è coinvolto e quali sono le sue informazioni di contatto? Quali sono le responsabilità di ciascuna parte interessata?
- Passaggi di risposta: in tutti i passaggi per la risposta agli incidenti, quali misure tattiche devono essere prese? Quali query deve eseguire l'analista? Quale codice va eseguito per ottenere il risultato desiderato?

  - Individuazione: come verrà individuato l'incidente?
  - Analisi: come verrà determinato l'ambito dell'impatto?
  - Contenimento: come verrà isolato l'incidente per limitarne la portata?
  - Sradicamento: come verrà rimossa la minaccia dall'ambiente?
  - Ripristino: in che modo il sistema o la risorsa interessati verranno riportati in produzione?
- Risultati previsto: dopo l'esecuzione delle query e del codice, qual è il risultato previsto del playbook?

Risorse

Best practice Well-Architected correlate:

- SEC10-BP02 Sviluppo di piani di gestione degli incidenti

Documenti correlati:

- Framework for Incident Response Playbooks
- Develop your own Incident Response Playbooks
- Incident Response Playbook Samples
- Building an AWS incident response runbook using Jupyter playbooks and CloudTrail Lake

SEC10-BP05 Preassegnazione dell'accesso

Verifica che le persone che intervengono dopo un incidente dispongano degli opportuni diritti di accesso allocati in AWS, così da ridurre i tempi necessari per l'analisi e il ripristino.

Anti-pattern comuni:

- Utilizzo dell'account root per la risposta agli incidenti.
- Modifica degli account utente esistenti.
- Manipolazione diretta delle autorizzazioni IAM quando si fornisce l'elevazione dei privilegi just-in-time.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

AWS raccomanda di ridurre o eliminare, ove possibile, la dipendenza da credenziali di lunga durata, a favore delle credenziali temporanee e dei meccanismi di escalation dei privilegi just-in-time. Le credenziali di lunga durata sono soggette a rischi per la sicurezza e aumentano il sovraccarico operativo. Per la maggior parte delle attività di gestione, nonché per quelle di risposta agli incidenti, si consiglia di implementare la federazione delle identità insieme all'escalation temporanea per l'accesso amministrativo. In questo modello, un utente richiede l'elevazione a un livello di privilegio superiore (come un ruolo di risposta agli incidenti) e, se è idoneo all'elevazione, la richiesta viene inviata al responsabile dell'approvazione. In caso di approvazione della richiesta, l'utente riceve una serie di credenziali AWS temporanee, utilizzabili per completare le proprie attività. Alla scadenza di tali credenziali, l'utente deve inviare una nuova richiesta di elevazione.

Si consiglia l'uso dell'escalation temporanea dei privilegi nella maggior parte degli scenari di risposta agli incidenti. Il modo corretto per eseguire questa operazione prevede l'utilizzo di AWS Security Token Service e policy di sessione per definire l'ambito dell'accesso.

Esistono scenari in cui le identità federate non sono disponibili, come nei seguenti casi:

- Interruzione correlata a un gestore dell'identità digitale (IdP) compromesso.
- Configurazione errata o errore umano che causa l'interruzione del sistema di gestione dell'accesso federato.
- Attività dannose, come un evento DDoS (Distributed Denial of Service) o l'indisponibilità del sistema.

Nei casi precedenti, occorre configurare l'accesso di emergenza break glass in modo da consentire l'indagine e la risoluzione tempestiva degli incidenti. È consigliabile ricorrere a utenti, gruppi o ruoli con le autorizzazioni opportune per l'esecuzione delle attività e l'accesso alle risorse AWS. Ricorri all'utente root solo per le attività che richiedono le credenziali dell'utente root. Per verificare che le persone che intervengono dopo un incidente dispongano del corretto livello di accesso ad AWS e ad altri sistemi pertinenti, ti consigliamo di eseguire la preallocazione di account dedicati. Gli account richiedono l'accesso con privilegi e devono essere rigorosamente controllati e monitorati. Gli account vanno creati con il minor numero di privilegi richiesti per eseguire le attività e il livello di accesso deve essere basato sui playbook inclusi nel piano di gestione degli incidenti.

Ricorri a utenti e ruoli specifici e dedicati come best practice. L'escalation temporanea dell'accesso di utenti o ruoli tramite l'aggiunta di policy IAM rende poco chiaro quale fosse l'accesso degli utenti durante l'incidente e si rischia la mancata revoca dei privilegi oggetto di escalation.

È importante rimuovere il maggior numero possibile di dipendenze per verificare che sia possibile ottenere l'accesso nel maggior numero possibile di scenari di errore. A supporto di ciò, crea un playbook per verificare che gli utenti di risposta agli incidenti vengano creati come utenti in un account di sicurezza dedicato e non gestiti tramite una federazione esistente o una soluzione di autenticazione Single Sign-On (SSO) Ogni singola persona che interviene dopo un incidente deve avere il proprio account denominato. La configurazione dell'account deve applicare una policy delle password complesse e l'autenticazione a più fattori (MFA). Se i playbook di risposta agli incidenti richiedono solo l'accesso al Console di gestione AWS, non è necessario che l'utente disponga di chiavi di accesso configurate né che sia esplicitamente autorizzato a creare chiavi di accesso. Questo può essere configurato con policy IAM o policy di controllo dei servizi come menzionato nelle best practice di sicurezza di AWS per le AWS Organizations SCP. Gli utenti non devono disporre di privilegi oltre alla capacità di assumere i ruoli di risposta agli incidenti in altri account.

Durante un incidente, potrebbe essere necessario concedere l'accesso ad altre persone interne o esterne per supportare le attività di analisi, correzione o ripristino. In questo caso, utilizza il meccanismo del playbook menzionato in precedenza e un processo per verificare la revoca immediata di qualsiasi accesso aggiuntivo immediatamente dopo la risoluzione dell'incidente.

Per verificare che l'uso dei ruoli di risposta agli incidenti possa essere adeguatamente monitorato e sottoposto ad audit, è essenziale che gli account IAM creati a tale scopo non siano condivisi tra le persone e che non si faccia ricorso all'Utente root dell'account AWS, salvo che non sia necessario per un'attività specifica. Se è richiesto l'utente root (ad esempio, l'accesso IAM a un account specifico non è disponibile), utilizza un processo separato con un playbook disponibile per verificare la disponibilità delle credenziali di accesso dell'utente root e del token MFA.

Per configurare le policy IAM per i ruoli di risposta agli incidenti, prendi in considerazione l'utilizzo di IAM Access Analyzer per generare policy basate su log AWS CloudTrail. In questo caso, concedi l'accesso come amministratore al ruolo di risposta agli incidenti per un account non di produzione e segui i playbook. Al termine, potrà essere creata una policy che consenta solo le azioni da intraprendere. Questa policy potrà quindi essere applicata a tutti i ruoli di risposta agli incidenti in tutti gli account. Puoi anche creare una policy IAM separata per ciascun playbook per semplificare gestione e audit. Esempi di playbook possono essere piani di risposta per ransomware, violazioni dei dati, perdita dell'accesso alla produzione e altri scenari.

Utilizza gli account di risposta agli incidenti per assumere i ruoli IAM dedicati di risposta agli incidenti in altri Account AWS. Questi ruoli devono essere configurati in modo che possano essere assunti solo dagli utenti nell'account di sicurezza e la relazione di trust deve richiedere che il principale chiamante sia autenticato tramite MFA. I ruoli devono utilizzare policy IAM con ambito limitato per

controllare l'accesso. Assicurati che tutte le richieste `AssumeRole` per questi ruoli vengano registrate in CloudTrail e notificate e che tutte le azioni intraprese utilizzando questi ruoli vengano registrate.

Ti consigliamo vivamente di denominare in modo chiaro gli account IAM e i ruoli IAM per trovarli facilmente nei log di CloudTrail. Un esempio potrebbe essere quello di denominare gli account IAM *<USER_ID>*`-BREAK-GLASS` e i ruoli IAM `BREAK-GLASS-ROLE`.

CloudTrail consente di creare log dell'attività delle API negli account AWS e va utilizzato per configurare gli avvisi sull'utilizzo dei ruoli di risposta agli incidenti. Fai riferimento al post del blog sulla configurazione degli avvisi quando vengono utilizzate le chiavi root. È possibile modificare le istruzioni in modo da configurare la metrica da filtro a filtro di Amazon CloudWatch sugli eventi `AssumeRole` relativi al ruolo IAM di risposta agli incidenti:

```
{ $.eventName = "AssumeRole" && $.requestParameters.roleArn =
 "<INCIDENT_RESPONSE_ROLE_ARN>" && $.userIdentity.invokedBy NOT EXISTS && $.eventType !
= "AwsServiceEvent" }
```

Vista la probabilità che i ruoli di risposta agli incidenti abbiano un livello di accesso elevato, è importante che questi avvisi vengano inviati a un gruppo ampio e gestiti tempestivamente.

Durante un incidente, è possibile che un membro del team di risposta richieda l'accesso a sistemi non direttamente protetti da IAM, ad esempio istanze Amazon Elastic Compute Cloud, database del servizio Amazon Relational Database o piattaforme Software-as-a-Service (SaaS). Si consiglia di utilizzare AWS Systems Manager Session Manager, anziché protocolli nativi come SSH o RDP per tutti gli accessi amministrativi alle istanze di Amazon EC2. Questo accesso può essere monitorato utilizzando IAM, che è sicuro e controllato. È inoltre possibile automatizzare parti dei playbook mediante i documenti AWS Systems Manager Run Command, in modo da ridurre gli errori degli utenti e migliorare i tempi di ripristino. Per l'accesso a database e strumenti di terze parti, ti consigliamo di archiviare le credenziali di accesso in Gestione dei segreti AWS e di concedere l'accesso ai ruoli delle persone che intervengono dopo gli incidenti.

Infine, la gestione degli account IAM per la risposta agli incidenti dovrebbe essere aggiunta ai processi degli utenti che si uniscono, si spostano o lasciano l'organizzazione e va riesaminata e testata periodicamente per verificare che sia consentito solo l'accesso previsto.

Risorse

Documenti correlati:

- Managing temporary elevated access to your AWS environment

- AWS Security Incident Response Guide
- AWS Elastic Disaster Recovery
- Strumento di gestione degli incidenti AWS Systems Manager
- Setting an account password policy for IAM users
- Using multi-factor authentication (MFA) in AWS
- Configurazione dell'accesso multi-account con MFA
- Utilizzo di IAM Access Analyzer per creare policy IAM
- Best Practices for AWS Organizations Service Control Policies in a Multi-Account Environment
- How to Receive Notifications When Your AWS Account's Root Access Keys Are Used
- Create fine-grained session permissions using IAM managed policies
- Break glass access

Video correlati:

- Automating Incident Response and Forensics in AWS
- DIY guide to runbooks, incident reports, and incident response
- Prepare for and respond to security incidents in your AWS environment

SEC10-BP06 Implementazione anticipata degli strumenti

Verifica che il team addetto alla sicurezza disponga degli strumenti giusti pre-implementati per ridurre i tempi di indagine fino al ripristino.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Per automatizzare le funzioni delle operazioni e la risposta di sicurezza, puoi utilizzare un set completo di API e strumenti AWS. Puoi automatizzare completamente le funzionalità di gestione delle identità, sicurezza della rete, protezione dei dati e monitoraggio e distribuirle utilizzando metodi di sviluppo software comuni già esistenti. Quando crei l'automazione della sicurezza, il sistema può monitorare, rivedere e avviare una risposta, anziché far sì che le persone monitorino la tua posizione di sicurezza e reagiscano manualmente agli eventi.

Se i team di risposta agli incidenti continuano a rispondere agli avvisi nello stesso modo, rischiano il cosiddetto affaticamento dagli avvisi ("alert fatigue"). Ciò significa che, nel corso del tempo, il team

può diventare desensibilizzato agli avvisi e commettere errori nella gestione di situazioni ordinarie o farsi sfuggire avvisi insoliti. L'automazione aiuta a evitare l'affaticamento dagli avvisi mediante funzioni che elaborano gli avvisi ripetitivi e ordinari, lasciando alle persone la gestione degli incidenti sensibili e univoci. L'integrazione di sistemi di rilevamento delle anomalie, come Amazon GuardDuty, AWS CloudTrail Insights e Amazon CloudWatch Anomaly Detection può ridurre l'impatto di avvisi frequenti basati su soglie.

Puoi migliorare i processi manuali automatizzando le fasi del processo a livello di programmazione. Dopo aver definito il modello di correzione di un evento, puoi scomporlo in una logica fruibile e scrivere il codice per eseguirla. Il team addetto alla risposta può quindi eseguire il codice per risolvere il problema. Nel corso del tempo, puoi automatizzare più fasi e, infine, gestire automaticamente intere classi di incidenti comuni.

Durante un'indagine di sicurezza, devi essere in grado di esaminare i log pertinenti per registrare e comprendere l'intera portata e la tempistica dell'incidente. I log servono anche per la generazione di avvisi, che indicano il verificarsi di determinate azioni di interesse. È fondamentale selezionare, attivare, memorizzare e impostare i meccanismi di query e recupero e impostare gli avvisi. Inoltre, una soluzione efficace per fornire gli strumenti di ricerca nei dati di log è Amazon Detective.

AWS offre oltre 200 servizi cloud e migliaia di funzionalità. Ti consigliamo di esaminare i servizi in grado di supportare e semplificare la tua strategia di risposta agli incidenti.

Oltre ai log, è necessario sviluppare e implementare una strategia di assegnazione tag. L'assegnazione dei tag può fornire il contesto per lo scopo di una risorsa AWS e può essere utilizzata anche per l'automazione.

Passaggi dell'implementazione

Seleziona e configura i log per analisi e avvisi

Consulta la seguente documentazione sulla configurazione dei log per la risposta agli incidenti:

- Logging strategies for security incident response
- SEC04-BP01 Configurazione dei log di servizi e applicazioni

Enable security services to support detection and response

AWS offre funzionalità investigative, preventive e reattive, nonché altri servizi utilizzabili per progettare soluzioni di sicurezza personalizzate. Per un elenco dei servizi più pertinenti per la risposta

agli incidenti di sicurezza, consulta Definizioni delle capacità del cloud e Homepage alle risposte agli incidenti di sicurezza.

Sviluppa e implementa una strategia di assegnazione tag

Ottenere informazioni contestuali sul caso d'uso aziendale e sulle parti interessante interne pertinenti relativi a una risorsa AWS può essere difficile. Un modo per farlo sono i tag che assegnano i metadati alle risorse AWS e sono composti da una chiave e un valore definiti dall'utente. Puoi creare i tag per classificare le risorse per scopo, proprietario, ambiente, tipo di dati elaborati e altri criteri di tua scelta.

Avere una strategia di assegnazione tag coerente può accelerare le risposta e ridurre al minimo il tempo dedicato al contesto organizzativo, consentendo di identificare e discernere rapidamente le informazioni contestuali su una risorsa AWS. I tag possono anche fungere da meccanismo per avviare le automazioni di risposta. Per maggiori dettagli su cosa taggare, consulta Taggare le risorse AWS. Dovrai prima definire i tag nella tua organizzazione e quindi implementare e applicare questa strategia di tag. Per maggiori dettagli su implementazione e applicazione, consulta Implement AWS resource tagging strategy using AWS Tag Policies and Service Control Policies (SCPs).

Risorse

Best practice Well-Architected correlate:

- SEC04-BP01 Configurazione dei log di servizi e applicazioni
- SEC04-BP02 Acquisizione di log, esiti e metriche in posizioni standardizzate

Documenti correlati:

- Logging strategies for security incident response
- Incident response cloud capability definitions

Esempi correlati:

- Threat Detection and Response with Amazon GuardDuty and Amazon Detective
- Workshop Security Hub
- Gestione delle vulnerabilità con Amazon Inspector

SEC10-BP07 Esecuzione di simulazioni

Man mano che le organizzazioni crescono e si evolvono nel tempo, aumentano anche le tipologie di minacce. Per questo motivo, è importante rivedere continuamente le capacità di risposta agli incidenti. L'esecuzione di simulazioni (note anche come giornate di gioco) è un metodo che può essere utilizzato per eseguire questa valutazione. Le simulazioni utilizzano scenari di eventi di sicurezza reali progettati per simulare le tattiche, le tecniche e le procedure (TTP) di un autore di minacce e consentire a un'organizzazione di esercitarsi e valutare le proprie capacità di risposta agli incidenti rispondendo a questi finti eventi informatici così come potrebbero verificarsi nella realtà.

Vantaggi dell'adozione di questa best practice: le simulazioni offrono una serie di vantaggi.

- Convalida della preparazione informatica e sviluppo della fiducia dei team di risposta agli incidenti.
- Verifica della precisione e dell'efficienza di strumenti e flussi di lavoro.
- Perfezionamento dei metodi di comunicazione ed escalation in linea con il piano di risposta agli incidenti.
- Opportunità di rispondere per i vettori meno comuni.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Esistono tre tipi principali di simulazioni:

- Simulazioni di situazioni di emergenza le simulazioni di situazioni di emergenza sono sessioni basate sulla discussione che coinvolgono le varie parti interessate alla risposta agli incidenti per mettere in pratica ruoli e responsabilità e utilizzare strumenti e playbook di comunicazione consolidati. Lo svolgimento dell'esercitazione può in genere essere eseguito in un'intera giornata in un luogo virtuale, in un luogo fisico o in una combinazione di questi tipi di luogo. Poiché è basato sulla discussione, questo tipo di esercitazione si concentra su processi, persone e collaborazione. La tecnologia è parte integrante della discussione, ma l'uso effettivo di strumenti o script di risposta agli incidenti in genere non rientra in questo tipo di simulazione.
- Esercitazioni con il team viola: questo tipo di esercitazioni aumenta il livello di collaborazione tra i team di risposta agli incidenti (team blu) e gli attori delle minacce simulate (team rosso). Il team blu è composto da membri del Security Operations Center (SOC), ma può includere anche altre parti interessate che sarebbero coinvolte durante un vero e proprio evento informatico. Il team rosso è composto da un team responsabile dei test di penetrazione o da parti interessate chiave esperte

in materia di sicurezza informatica. Il team rosso lavora assieme ai coordinatori dell'esercitazione durante la progettazione di uno scenario in modo che questi sia accurato e fattibile. Durante le esercitazioni del team viola, l'attenzione è rivolta principalmente ai meccanismi di rilevamento, agli strumenti e alle procedure operative standard (SOP) a supporto della risposta agli incidenti.

- Esercitazioni con il team rosso: durante un'esercitazione con il team rosso, l'attacco (team rosso) effettua una simulazione per raggiungere un determinato obiettivo o una serie di obiettivi da un ambito predeterminato. I difensori (team blu) non saranno necessariamente a conoscenza della portata e della durata dell'esercitazione, il che fornisce una valutazione più realistica di come risponderebbero a un incidente reale. Poiché le esercitazioni con il team rosso possono basarsi su test invasivi, procedi con cautela e implementa controlli per verificare che l'esercitazione non causi danni effettivi all'ambiente.

Prendi in considerazione la possibilità di svolgere simulazioni informatiche a intervalli regolari. Ogni tipo di esercitazione può offrire vantaggi unici ai partecipanti e all'organizzazione nel suo insieme; potresti, quindi, scegliere di iniziare con tipi di simulazione meno complessi (come le simulazioni di situazioni di emergenza) e passare a tipi di simulazione più complessi (esercitazioni del team rosso). È necessario selezionare un tipo di simulazione in base alla maturità, alle risorse e ai risultati desiderati a livello di sicurezza. Alcuni clienti potrebbero scegliere di non eseguire le esercitazioni del team rosso a causa della loro complessità e dei loro costi.

Passaggi dell'implementazione

Indipendentemente dal tipo di simulazione scelto, le simulazioni sono in genere caratterizzate dai seguenti passaggi di implementazione:

1. Definisci gli elementi principali dell'esercitazione: definisci scenario e obiettivi della simulazione. Lo scenario e gli obiettivi dovrebbero essere entrambi accettati dalla leadership.
2. Identifica le parti interessate principali: come minimo, un'esercitazione prevede la presenza di coordinatori e partecipanti. A seconda dello scenario, potrebbero essere coinvolte altre parti interessate come la leadership legale, delle comunicazioni o esecutiva.
3. Crea ed esegui il test dello scenario: potrebbe essere necessario ridefinire lo scenario durante la creazione se risulta impossibile implementare elementi specifici. Come risultato di questa fase è previsto uno scenario definitivo.
4. Fai svolgere la simulazione: il tipo di simulazione determina il tipo di svolgimento usato (uno scenario basato su supporto cartaceo o uno scenario con simulazione altamente tecnologica). I coordinatori dovrebbero allineare le loro tattiche di svolgimento agli oggetti dell'esercitazione e dovrebbero coinvolgere tutti i partecipanti ove possibile per ottimizzare i benefici.

5. Predisponi il report post-azione (AAR): identifica le aree positive, quelle da migliorare e le potenziali lacune. Il report AAR dovrebbe misurare l'efficacia della simulazione e la risposta del team all'evento simulato in modo che i progressi possano essere monitorati nel tempo con simulazioni future.

Risorse

Documenti correlati:

- Guida sulla risposta agli incidenti di sicurezza di AWS

Video correlati:

- AWS GameDay - Security Edition
- Esecuzione di simulazioni di risposta agli incidenti di sicurezza efficaci

SEC10-BP08 Definizione di un framework per apprendere dagli incidenti

L'implementazione di un framework basato sulle lezioni apprese e di una capacità di analisi delle cause principali non solo contribuisce a migliorare le capacità di risposta agli incidenti, ma aiuta anche a prevenire il ripetersi dell'incidente. Imparando da ogni incidente, puoi evitare di ripetere gli errori, i rischi o le configurazioni non valide, non solo migliorando il tuo livello di sicurezza, ma anche riducendo al minimo il tempo speso in situazioni evitabili.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

È importante implementare un framework basato sulle lezioni apprese in grado di stabilire e raggiungere, a un livello elevato, i seguenti punti:

- Quando si tiene un framework basato sulle lezioni apprese?
- Cosa comporta il processo basato sulle lezioni apprese?
- Come viene eseguito un framework basato sulle lezioni apprese?
- Chi è coinvolto nel processo e in che modo?
- Come vengono identificate le aree di miglioramento?
- In che modo garantisci che i miglioramenti vengano monitorati e implementati in modo efficace?

Il framework non deve concentrarsi sugli individui, ma sul miglioramento di strumenti e processi.

Passaggi dell'implementazione

A parte i risultati di alto livello sopra elencati, è importante porsi le domande giuste per trarre il massimo valore (informazioni che portano a miglioramenti attuabili) dal processo. Considera queste domande per iniziare a promuovere le discussioni sulle lezioni apprese:

- Qual è stato l'incidente?
- Quando è stato identificato per la prima volta l'incidente?
- Come è stato identificato?
- Quali sistemi hanno avvisato dell'attività?
- Quali sistemi, servizi e dati sono stati coinvolti?
- Cosa è successo nello specifico?
- Cosa ha funzionato bene?
- Cosa non ha funzionato bene?
- Quale processo o quali procedure non sono riusciti a scalare per rispondere all'incidente?
- Cosa può essere migliorato nelle seguenti aree:
  - Persone
    - Le persone da contattare erano effettivamente disponibili e l'elenco dei contatti era aggiornato?
    - Le persone presentavano lacune nella formazione o nelle capacità necessarie per rispondere e indagare efficacemente sull'incidente?
    - Le risorse appropriate erano pronte e disponibili?
  - Processo
    - Sono stati seguiti i processi e le procedure?
    - I processi e le procedure erano documentati e disponibili per questo tipo di incidente?
    - Mancavano i processi e le procedure richiesti?
    - Il team di risposta è stato in grado di accedere tempestivamente alle informazioni necessarie per rispondere al problema?
  - Tecnologia
    - I sistemi di avviso esistenti hanno identificato e segnalato efficacemente l'attività?
    - Come si sarebbe potuto ridurre il tempo di rilevamento del 50%?

- Gli avvisi esistenti devono essere migliorati o è necessario creare nuovi avvisi per questo (tipo di) incidente?
- Gli strumenti esistenti hanno consentito un'indagine efficace (ricerca/analisi) dell'incidente?
- Cosa si può fare per identificare prima questo tipo di incidente?
- Cosa si può fare per evitare che questo tipo di incidente si ripeta?
- A chi appartiene il piano di miglioramento e come verifichi che sia stato implementato?
- Qual è la tempistica per l'implementazione e il test del monitoraggio aggiuntivo o dei controlli e dei processi preventivi?

Questo elenco non è esaustivo, ma può fungere da punto di partenza per individuare quali sono le esigenze dell'organizzazione e dell'attività e come analizzarle per imparare in modo più efficace dagli incidenti e migliorare costantemente il proprio livello di sicurezza. La cosa più importante è iniziare incorporando le lezioni apprese come parte standard del processo di risposta agli incidenti, della documentazione e delle aspettative di tutti le parti interessate.

Risorse

Documenti correlati:

- AWS Security Incident Response Guide - Establish a framework for learning from incidents
- NCSC CAF guidance - Lessons learned

Sicurezza delle applicazioni

Domanda

- SEC 11. Come si incorporano e convalidano le proprietà di sicurezza delle applicazioni nell'intero ciclo di vita di progettazione, sviluppo e implementazione?

SEC 11. Come si incorporano e convalidano le proprietà di sicurezza delle applicazioni nell'intero ciclo di vita di progettazione, sviluppo e implementazione?

La formazione del personale, l'esecuzione di test tramite automazione, l'identificazione delle dipendenze e la convalida delle proprietà di sicurezza di strumenti e applicazioni riducono la probabilità del verificarsi di problemi di sicurezza nei carichi di lavoro di produzione.

Best practice

- SEC11-BP01 Formazione per la sicurezza delle applicazioni
- SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test
- SEC11-BP03 Esecuzione di test di penetrazione a intervalli regolari
- SEC11-BP04 Esecuzione di revisioni del codice
- SEC11-BP05 Centralizzazione dei servizi per pacchetti e dipendenze
- SEC11-BP06 Implementazione programmatica del software
- SEC11-BP07 Valutazione regolare delle proprietà di sicurezza delle pipeline
- SEC11-BP08 Creazione di un programma per l'integrazione della titolarità della sicurezza nei team responsabili del carico di lavoro

SEC11-BP01 Formazione per la sicurezza delle applicazioni

Offri al tuo team una formazione su pratiche operative e di sviluppo sicure per consentire la creazione di software sicuro e di alta qualità. In questo modo, il team può a prevenire, rilevare e correggere i problemi di sicurezza nelle prime fasi del ciclo di vita dello sviluppo. Valuta la possibilità di fornire una formazione su modellazione delle minacce, pratiche di codifica sicure e utilizzo di servizi per configurazioni e operazioni sicure. Dai la possibilità al team di accedere alla formazione tramite risorse self-service e raccogli regolarmente i feedback per garantire un miglioramento continuo.

Risultato desiderato: il tuo team dispone delle conoscenze e delle competenze necessarie per progettare e creare software pensando alla sicurezza fin dal principio. Grazie alla formazione su modellazione delle minacce e pratiche di sviluppo sicure, il team ottiene una conoscenza approfondita dei potenziali rischi per la sicurezza e dei metodi per mitigarli durante il ciclo di vita dello sviluppo software (SDLC). Questo approccio proattivo alla sicurezza si integra nella cultura del tuo team e hai la possibilità di identificare e correggere tempestivamente potenziali problemi di sicurezza. Di conseguenza, il tuo team crea software e funzionalità sicuri e di alta qualità in modo più efficiente, accelerando così le tempistiche di consegna complessive. All'interno dell'organizzazione la cultura della sicurezza è collaborativa e inclusiva: la titolarità della sicurezza è condivisa tra tutti gli sviluppatori.

Anti-pattern comuni:

- Attendi una revisione della sicurezza e poi valuti le proprietà di sicurezza di un sistema.
- Assegni tutte le decisioni in materia di sicurezza a un team responsabile della sicurezza.
- Manca la comunicazione della correlazione tra le decisioni adottate durante il ciclo di vita dello sviluppo software e le aspettative o policy complessive dell'organizzazione.

- Svolgi il processo di revisione della sicurezza in una fase troppo tardiva.

Vantaggi dell'adozione di questa best practice:

- Migliore identificazione dei requisiti aziendali per la sicurezza all'inizio del ciclo di sviluppo.
- Capacità di identificare e correggere più rapidamente possibili problemi di sicurezza, per una distribuzione più rapida delle funzionalità.
- Migliore qualità del software e dei sistemi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Per creare software sicuro e di alta qualità, offri al tuo team una formazione sulle pratiche comuni per lo sviluppo e la gestione delle applicazioni in sicurezza. In questo modo, il team può a prevenire, rilevare e correggere i problemi di sicurezza nelle prime fasi del ciclo di vita dello sviluppo, accelerando così le tempistiche di consegna.

Per raggiungere questo obiettivo, valuta la possibilità di formare il tuo team sulla modellazione delle minacce utilizzando risorse AWS come il workshop sulla modellazione delle minacce. La modellazione delle minacce può aiutare il team a comprendere i potenziali rischi per la sicurezza e a progettare i sistemi tenendo conto della sicurezza fin dal principio. Inoltre, puoi fornire l'accesso alle risorse di formazione di AWS Training Certification, di settore o dei Partner AWS sulle pratiche di sviluppo sicure. Per maggiori dettagli su un approccio completo alla progettazione, allo sviluppo, alla protezione e alla gestione efficiente su larga scala, consulta AWS DevOps Guidance.

Definisci e comunica in modo chiaro il processo di revisione della sicurezza dell'organizzazione e descrivi le responsabilità del tuo team, del team addetto alla sicurezza e delle altre parti interessate Pubblica linee guida self-service, esempi di codice e modelli che mostrino come soddisfare i requisiti di sicurezza. Puoi utilizzare servizi AWS come AWS CloudFormation, AWS Cloud Development Kit (AWS CDK) (AWS CDK) Constructs e Catalogo dei servizi per fornire configurazioni sicure preapprovate e ridurre la necessità di configurazioni personalizzate.

Raccogli periodicamente dal tuo team feedback sull'esperienza con il processo di revisione della sicurezza e la formazione correlata, e usalo per ottenere un miglioramento continuo. Organizza GameDay o campagne di bug bash per identificare e risolvere i problemi di sicurezza, rafforzando al contempo le competenze del tuo team.

Passaggi dell'implementazione

1. Identifica le esigenze di formazione: valuta l'attuale livello delle competenze e le lacune nelle conoscenze all'interno del team in materia di pratiche di sviluppo sicure attraverso sondaggi, revisioni del codice o discussioni con i membri del team.
2. Pianifica la formazione: in base alle esigenze identificate, crea un piano di formazione che copra argomenti rilevanti come modellazione delle minacce, pratiche di codifica sicure, test di sicurezza e pratiche di implementazione sicure. Utilizza risorse come il workshop sulla modellazione delle minacce e i programmi di formazione di AWS Training and Certification, di settore o dei Partner AWS.
3. Pianifica e offri corsi di formazione: pianifica sessioni di formazione o workshop periodici per il tuo team. Possono essere tenuti da un istruttore o personalizzati, a seconda delle preferenze e della disponibilità del team. Incoraggia lo svolgimento di esercizi ed esempi pratici per rafforzare l'apprendimento.
4. Definisci un processo di revisione della sicurezza: collabora con il tuo team addetto alla sicurezza e con le altre parti interessate per definire chiaramente il processo di revisione della sicurezza per le tue applicazioni. Documenta le responsabilità di ogni team o individuo coinvolto nel processo, inclusi i team addetti allo sviluppo e alla sicurezza e incluse eventuali altre parti interessate.
5. Crea risorse self-service: sviluppa linee guida self-service, esempi di codice e modelli che mostrino come soddisfare i requisiti di sicurezza dell'organizzazione. Valuta la possibilità di utilizzare servizi AWS come CloudFormation, AWS CDK Constructs e Catalogo dei servizi per fornire configurazioni sicure preapprovate e ridurre la necessità di configurazioni personalizzate.
6. Comunica e socializza: comunica in modo efficace al tuo team il processo di revisione della sicurezza e le risorse self-service disponibili. Conduci sessioni di formazione o workshop per far acquisire familiarità con queste risorse e per verificare che sappiano come usarle.
7. Raccogli i feedback e migliora i processi: raccogli periodicamente dal tuo team feedback sull'esperienza con il processo di revisione di sicurezza e la formazione correlata. Utilizza i feedback per identificare le aree di miglioramento e migliorare continuamente i materiali di formazione, le risorse self-service e il processo di revisione della sicurezza.
8. Svolgi esercizi di sicurezza: organizza GameDay o campagne di bug bash per identificare e risolvere i problemi di sicurezza all'interno delle applicazioni. Questi esercizi non solo aiutano a scoprire potenziali vulnerabilità, ma offrono anche opportunità pratiche di apprendimento per il team, volte a migliorare le competenze in materia di sviluppo e gestione in sicurezza.
9. Continua a imparare e migliorare: incoraggia il tuo team a rimanere aggiornato sulle pratiche, sugli strumenti e sulle tecniche di sviluppo in sicurezza più recenti. Rivedi e aggiorna regolarmente i

materiali e le risorse di formazione per riflettere le best practice e il panorama della sicurezza in continua evoluzione.

Risorse

Best practice correlate:

- SEC11-BP08 Creazione di un programma per l'integrazione della titolarità della sicurezza nei team responsabili del carico di lavoro

Documenti correlati:

- AWS Training and Certification
- How to think about cloud security governance
- How to approach threat modeling
- Accelerating training – The AWS Skills Guild
- AWS DevOps Sagas

Video correlati:

- Proactive security: Considerations and approaches

Esempi correlati:

- Workshop on threat modeling
- Industry awareness for developers

Servizi correlati:

- AWS CloudFormation
- AWS Cloud Development Kit (AWS CDK) (AWS CDK) Costrutti di
- Service Catalog

SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test

Automatizza i test per le proprietà di sicurezza lungo il ciclo di vita di sviluppo e test. L'automazione semplifica l'identificazione coerente e ripetibile dei potenziali problemi nel software prima del rilascio, riducendo il rischio di riscontrare problemi di sicurezza nel software fornito.

Risultato desiderato: l'obiettivo dei test automatizzati è fornire una soluzione programmatica per l'individuazione di potenziali problemi nelle fasi iniziali e spesso durante l'intero ciclo di vita dello sviluppo. Automatizzando i test di regressione, puoi ripetere l'esecuzione di test funzionali e non funzionali per verificare che il software testato in precedenza continui ad avere le prestazioni previste dopo una modifica. Quando definisci test di unità di sicurezza per verificare la presenza di configurazioni errate comuni, come autorizzazioni non corrette o mancanti, puoi identificare e correggere i problemi all'inizio del processo di sviluppo.

Per l'automazione dei test vengono usati casi di test dedicati per la convalida delle applicazioni, in base ai requisiti e alle funzionalità desiderate. Il risultato dei test automatici è basato sul confronto dell'output del test generato con quello previsto, che accelera l'intero ciclo di vita dei test. Metodologie di test come i test di regressione e le suite di test di unità sono ideali per l'automazione. L'automazione dei test delle proprietà di sicurezza permette agli sviluppatori di ricevere in automatico feedback senza attendere una revisione della sicurezza. I test automatici sotto forma di analisi statica o dinamica del codice possono migliorare la qualità del codice e semplificare il rilevamento dei potenziali problemi software all'inizio del ciclo di vita di sviluppo.

Anti-pattern comuni:

- Mancata comunicazione dei casi di test e dei risultati dei test automatici.
- Esecuzione dei test solo immediatamente prima di un rilascio.
- Automazione dei casi di test con requisiti che cambiano spesso.
- Assenza di linee guida su come gestire i risultati dei test di sicurezza.

Vantaggi dell'adozione di questa best practice:

- Riduzione della dipendenza da valutazioni personali delle proprietà di sicurezza dei sistemi.
- Migliore coerenza grazie a esiti uniformi tra più flussi di lavoro.
- Minore probabilità di introdurre problemi di sicurezza nel software di produzione.
- Intervallo di tempo più breve tra l'individuazione e la correzione grazie all'identificazione più tempestiva dei problemi software.

- Maggiore visibilità su comportamenti sistematici o ripetuti tra più flussi di lavoro, utile per favorire miglioramenti in tutta l'organizzazione.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Durante lo sviluppo del software, adotta diversi meccanismi di test in modo da avere la certezza di testare l'applicazione per requisiti funzionali, basati sulla logica di business, e non funzionali, incentrati sull'affidabilità, sulle prestazioni e sulla sicurezza dell'applicazione.

I test di sicurezza statici dell'applicazione analizzano il codice sorgente in cerca di modelli di sicurezza anomali e forniscono indicazioni su un codice soggetto a errori. I test di sicurezza statici dell'applicazione si basano su input statici, come la documentazione (definizione dei requisiti, documentazione sulla progettazione e specifiche di progettazione) e il codice sorgente dell'applicazione, per testare un'ampia gamma di problemi di sicurezza noti. Gli analizzatori di codice statici possono contribuire ad accelerare l'analisi di volumi elevati di codice. Il NIST Quality Group fornisce un confronto tra gli analizzatori della sicurezza del codice sorgente, che include strumenti open source per la scansione del codice byte e la scansione del codice binario.

Integra i test statici con metodologie di test della sicurezza tramite analisi dinamica, che eseguono test sull'applicazione in esecuzione per identificare potenziali comportamenti imprevisti. I test dinamici consentono di individuare potenziali problemi non rilevabili tramite l'analisi statica. L'esecuzione di test nelle fasi di repository, compilazione e pipeline del codice permette di verificare potenziali problemi di tipi diversi, evitandone la presenza nel codice. Amazon Q Developer fornisce suggerimenti sul codice, tra cui la scansione di sicurezza, nell'ambiente IDE del generatore. La Sicurezza di Amazon CodeGuru è in grado di identificare problemi critici, problemi di sicurezza e bug difficili da individuare durante lo sviluppo di applicazioni e fornisce consigli per migliorare la qualità del codice. L'estrazione di documenti SBOM (Software Bill of Material) consente anche di estrarre un record formale contenente i dettagli e le relazioni dei vari componenti utilizzati nella creazione del software. Ciò consente di gestire le vulnerabilità in modo informato e di identificare rapidamente le dipendenze tra software o componenti e i rischi legati alla catena di approvvigionamento.

Il workshop Security for Developers utilizza strumenti AWS per gli sviluppatori, come AWS CodeBuild, AWS CodeCommit e AWS CodePipeline, per l'automazione della pipeline di rilascio che comprendono metodologie di test SAST e DAST.

Lungo il ciclo di vita di sviluppo del software definisci un processo iterativo che includa revisioni periodiche dell'applicazione con il team responsabile della sicurezza. Il feedback raccolto da

queste revisioni della sicurezza deve essere affrontato e convalidato come parte della revisione dell'idoneità per il rilascio. Queste revisioni permettono di stabilire una solida posizione di sicurezza per l'applicazione e forniscono agli sviluppatori feedback di utilità pratica per affrontare i potenziali problemi.

Passaggi dell'implementazione

- Implementa un ambiente IDE, una revisione del codice e strumenti CI/CD coerenti che includano test di sicurezza.
- Determina le fasi del ciclo di vita di sviluppo del software in cui è opportuno bloccare le pipeline anziché informare semplicemente gli sviluppatori riguardo alla necessità di risolvere i problemi.
- Automated Security Helper (ASH) è un esempio di strumento di scansione open source che aiuta a verificare la sicurezza del codice.
- L'esecuzione di test o analisi del codice mediante strumenti automatizzati, come Amazon Q Developer, integrato con gli ambienti IDE per sviluppatori, e la Sicurezza di Amazon CodeGuru per la scansione del codice al momento del commit, consente agli sviluppatori di ricevere feedback al momento giusto.
- Se sviluppi usando AWS Lambda, puoi sfruttare Amazon Inspector per la scansione del codice dell'applicazione nelle tue funzioni.
- Se le pipeline CI/CD includono test automatici, devi usare un sistema di gestione dei ticket per tenere traccia della notifica e della correzione dei problemi software.
- Per test di sicurezza che possono generare esiti, il collegamento a linee guida per la correzione permette agli sviluppatori di migliorare la qualità del codice.
- Analizza regolarmente gli esiti ottenuti dagli strumenti automatici per definire le priorità delle successive iniziative di automazione, formazione degli sviluppatori o creazione di campagne di sensibilizzazione.
- Per estrarre documenti SBOM nell'ambito delle pipeline CI/CD, usa Amazon Inspector SBOM Generator per creare SBOM per archivi, immagini di container, directory, sistemi locali e binari Go e Rust compilati nel formato CycloneDX SBOM.

Risorse

Best practice correlate:

- DevOps Guidance: DL.CR.3 Establish clear completion criteria for code tasks

Documenti correlati:

- Distribuzione e implementazione continue

- AWS Partner con competenze in DevOps
- AWS Security Competency Partners per la sicurezza delle applicazioni
- Choosing a Well-Architected CI/CD approach
- Secrets detection in Amazon CodeGuru Security
- Amazon CodeGuru Security Detection Library
- Accelerate deployments on AWS with effective governance
- How AWS approaches automating safe, hands-off deployments
- How Amazon CodeGuru Security helps you effectively balance security and velocity

Video correlati:

- Hands-off: Automating continuous delivery pipelines at Amazon
- Automating cross-account CI/CD pipelines
- The Software Development Process at Amazon
- Testing software and systems at Amazon

Esempi correlati:

- Industry awareness for developers
- Automated Security Helper (ASH)
- AWS CodePipeline Governance - Github

SEC11-BP03 Esecuzione di test di penetrazione a intervalli regolari

Esegui regolarmente test di penetrazione sul software. Questo meccanismo ti consente di identificare potenziali problemi relativi al software che non possono essere rilevati dai test automatizzati o dalla revisione manuale del codice e può anche aiutarti a capire l'efficacia dei tuoi controlli di rilevamento. I test di penetrazione devono determinare se il software può essere reso operativo in modi imprevisti, ad esempio esponendo dati che da proteggere o concedendo autorizzazioni più elevate del previsto.

Risultato desiderato: utilizzo del test di penetrazione per rilevare, correggere e convalidare le proprietà di sicurezza dell'applicazione. È necessario eseguire test di penetrazione regolari e pianificati nell'ambito del ciclo di vita di sviluppo del software. Gli esiti ottenuti dai test di penetrazione devono essere gestiti prima del rilascio del software. Devi analizzare gli esiti dei test di penetrazione per identificare l'eventuale presenza di problemi identificabili con l'automazione. Un processo di esecuzione di test di penetrazione regolare e ripetibile, con un meccanismo di feedback attivo, aiuta a stabilire linee guida per gli sviluppatori e migliora la qualità del software.

Anti-pattern comuni:

- Esecuzione di test di penetrazione solo per problemi di sicurezza noti o comuni.
- Esecuzione di test di penetrazione delle applicazioni senza gli strumenti e le librerie di terze parti dipendenti.
- Esecuzione di test di penetrazione solo per i problemi di sicurezza relativi ai pacchetti, senza valutare la logica di business implementata.

Vantaggi dell'adozione di questa best practice:

- Maggiore certezza riguardo alle proprietà di sicurezza del software prima del rilascio.
- Opportunità di identificare i modelli comportamentali preferiti delle applicazioni, per una migliore qualità del software.
- Presenza di un ciclo di feedback che identifica all'inizio del ciclo di sviluppo i punti in cui l'automazione o una formazione aggiuntiva possono migliorare le proprietà di sicurezza del software.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I test di penetrazione sono un esercizio strutturato per l'esecuzione di test di sicurezza in cui vengono eseguiti scenari di violazione della sicurezza pianificati per rilevare, correggere e convalidare i controlli di sicurezza. I test di penetrazione partono dalla ricognizione, durante la quale si raccolgono dati in base all'attuale progettazione dell'applicazione e alle sue dipendenze. Viene creato ed eseguito un elenco selezionato di scenari di test specifici per la sicurezza. Lo scopo principale di questi test è rivelare i problemi di sicurezza nell'applicazione che potrebbero essere sfruttati per ottenere l'accesso indesiderato all'ambiente o l'accesso non autorizzato ai dati. Devi eseguire test

di penetrazione quando lanci nuove funzionalità o ogni volta che l'applicazione viene sottoposta a modifiche importanti durante l'implementazione tecnica o di funzioni.

Devi identificare la fase più appropriata del ciclo di vita di sviluppo in cui eseguire i test di penetrazione. Questi test devono essere eseguiti nelle fasi finali, in modo che la funzionalità del sistema sia vicina allo stato di rilascio previsto, ma con tempo sufficiente per la correzione di eventuali problemi.

Passaggi dell'implementazione

- Adotta un processo strutturato per definire l'ambito dei test di penetrazione. Basare il processo sul modello di minaccia costituisce una buona soluzione per mantenere il contesto.
- Identifica la fase più appropriata del ciclo di vita di sviluppo in cui eseguire test di penetrazione. Questi devono avvenire quando sono previste modifiche minime nell'applicazione, ma quando vi è ancora tempo sufficiente per apportare eventuali correzioni.
- Prepara gli sviluppatori su cosa aspettarsi dagli esiti dei test di penetrazione e su come ottenere informazioni sulla correzione.
- Usa strumenti per accelerare il processo di esecuzione dei test di penetrazione automatizzando test comuni o ripetibili.
- Analizza gli esiti dei test di penetrazione per identificare problemi di sicurezza sistematici e usa questi dati per definire altri test automatici e formazione continua per gli sviluppatori.

Risorse

Best practice correlate:

- SEC11-BP01 Formazione per la sicurezza delle applicazioni
- SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test

Documenti correlati:

- La pagina relativa ai test di penetrazione AWS fornisce una guida dettagliata per i test di penetrazione su AWS
- Accelerate deployments on AWS with effective governance
- AWS Security Competency Partners
- Modernize your penetration testing architecture on AWS Fargate

- AWS Fault Injection Simulator

Esempi correlati:

- Automazione dei test API con AWS CodePipeline (GitHub)
- Helper di sicurezza automatizzato (GitHub)

SEC11-BP04 Esecuzione di revisioni del codice

Implementa le revisioni del codice per verificare la qualità e la sicurezza del software in fase di sviluppo. Le revisioni del codice prevedono che membri del team diversi da quelli che hanno originariamente scritto il codice esaminino il codice stesso per individuare potenziali problemi e vulnerabilità e l'aderenza agli standard e alle best practice in materia di codifica. Questo processo aiuta a individuare errori, incongruenze e difetti di sicurezza che potrebbero essere stati trascurati dallo sviluppatore originale. Utilizza strumenti automatici per facilitare la revisione del codice.

Risultato desiderato: le revisioni del codice vengono incluse durante la fase di sviluppo per aumentare la qualità del software in fase di scrittura. Le competenze dei membri meno esperti del team migliorano grazie ad apprendimenti identificati durante la revisione del codice. Vengono identificate le opportunità di automazione e il processo di revisione del codice viene supportato con strumenti e test automatizzati.

Anti-pattern comuni:

- Il codice non viene revisionato prima dell'implementazione.
- Scrittura e revisione del codice effettuate dalla stessa persona.
- Mancato utilizzo dell'automazione e degli strumenti per facilitare o orchestrare le revisioni del codice.
- Mancata formazione degli sviluppatori sulla sicurezza dell'applicazione prima di eseguire la revisione del codice.

Vantaggi dell'adozione di questa best practice:

- Migliore qualità del codice.
- Maggiore coerenza dello sviluppo del codice attraverso il riutilizzo di approcci comuni.
- Riduzione del numero di problemi riscontrati durante i test di penetrazione e nelle fasi successive.

- Migliore circolazione delle informazioni all'interno del team.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Le revisioni del codice aiutano a verificare la qualità e la sicurezza del software durante la fase di sviluppo. Le revisioni manuali prevedono che membri del team diversi da quelli che hanno originariamente scritto il codice esaminino il codice stesso per individuare potenziali problemi e vulnerabilità e l'aderenza agli standard e alle best practice in materia di codifica. Questo processo aiuta a individuare errori, incongruenze e difetti di sicurezza che potrebbero essere stati trascurati dallo sviluppatore originale.

Valuta la possibilità di utilizzare Sicurezza di Amazon CodeGuru per condurre revisioni automatiche del codice. Sicurezza di CodeGuru utilizza il machine learning e il ragionamento automatico per analizzare il codice e identificare potenziali vulnerabilità di sicurezza e problemi di codifica. Integra le revisioni automatiche del codice con i repository di codice e le pipeline di integrazione continua/distribuzione continua (CI/CD) esistenti.

Passaggi dell'implementazione

1. Stabilisci un processo di revisione del codice:
   - Definisci quando devono essere eseguite le revisioni del codice, ad esempio prima di unire il codice nel ramo principale o prima dell'implementazione in produzione.
   - Determina chi deve essere coinvolto nel processo di revisione del codice, ad esempio i membri del team, gli sviluppatori senior e gli esperti di sicurezza.
   - Decidi la metodologia di revisione del codice, inclusi il processo e gli strumenti da utilizzare.
2. Configura gli strumenti di revisione del codice:
   - Valuta e seleziona gli strumenti di revisione del codice più adatti alle esigenze del tuo team, come le richieste pull di GitHub o Sicurezza di CodeGuru.
   - Integra gli strumenti scelti con i tuoi repository di codice e le pipeline CI/CD esistenti.
   - Configura gli strumenti per applicare i requisiti di revisione del codice, come il numero minimo di revisori e le regole di approvazione.
3. Definisci checklist e linee guida per la revisione del codice:
   - Crea una checklist o elabora delle linee guida per la revisione del codice che descrivano gli elementi da esaminare. Prendi in considerazione fattori come la qualità del codice, le vulnerabilità di sicurezza, l'aderenza agli standard di codifica e le prestazioni.

- Condividi la checklist o le linee guida con il team di sviluppo e verifica che tutti comprendano le aspettative.

4. Forma gli sviluppatori sulle best practice per la revisione del codice:
   - Offri una formazione al tuo team su come condurre revisioni del codice efficaci.
   - Educa il team in merito ai principi di sicurezza delle applicazioni e alle vulnerabilità comuni da individuare durante le revisioni.
   - Incoraggia la condivisione delle conoscenze e abbina sessioni di programmazione per migliorare le competenze dei membri del team meno esperti.
5. Implementa il processo di revisione del codice:
   - Integra la fase di revisione del codice nel flusso di lavoro di sviluppo, ad esempio creando una richiesta pull e assegnando i revisori.
   - Richiedi che le modifiche al codice siano sottoposte a una revisione del codice prima dell'unione o dell'implementazione.
   - Incoraggia una comunicazione aperta e la comunicazione di feedback costruttivi durante il processo di revisione.
6. Monitora e migliora i processi:
   - Verifica regolarmente l'efficacia del processo di revisione del codice e raccogli feedback dal team.
   - Identifica le opportunità di automazione o di miglioramento degli strumenti per semplificare il processo di revisione del codice.
   - Aggiorna e perfeziona continuamente la checklist o le linee guida per la revisione del codice in base agli apprendimenti e alle best practice di settore.
7. Sensibilizza sull'importanza della revisione del codice:
   - Sottolinea l'importanza delle revisioni del codice per mantenere la qualità e la sicurezza del codice a un livello elevato.
   - Celebra i successi e gli apprendimenti frutto del processo di revisione del codice.
   - Incoraggia lo sviluppo di un ambiente collaborativo e di supporto in cui gli sviluppatori si sentano a proprio agio nel fornire e ricevere feedback.

Risorse

Best practice correlate:

- SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test

Documenti correlati:

- DevOps Guidance: DL.CR.2 Perform peer review for code changes
- About pull requests in GitHub

Esempi correlati:

- Automate code reviews with Amazon CodeGuru Security
- Automating detection of security vulnerabilities and bugs in CI/CD pipelines using Amazon CodeGuru Security CLI

Video correlati:

- Continuous improvement of code quality with Amazon CodeGuru Security

SEC11-BP05 Centralizzazione dei servizi per pacchetti e dipendenze

Fornisci servizi centralizzati per permettere ai tuoi team di ottenere pacchetti software e altre dipendenze. Questo approccio permette la convalida dei pacchetti prima di includerli nel software scritto e fornisce un'origine dati per l'analisi del software usato nell'organizzazione.

Risultato desiderato: il carico di lavoro viene creato sulla base di pacchetti software esterni in aggiunta al codice scritto dal tuo team. In questo modo, è più facile implementare funzionalità usate ripetutamente, come un parser JSON o una libreria di crittografia. Le origini per tali pacchetti e dipendenze vengono centralizzate, così che il tuo team addetto alla sicurezza possa convalidarle prima che vengano utilizzate. Questo approccio viene utilizzato insieme ai flussi di test manuali e automatici per garantire ulteriormente la qualità del software sviluppato.

Anti-pattern comuni:

- Recupero di pacchetti da repository arbitrari su Internet.
- Mancata esecuzione di test sui nuovi pacchetti prima di renderli disponibili agli sviluppatori.

Vantaggi dell'adozione di questa best practice:

- Migliore comprensione dei pacchetti usati nel software sviluppato.

- Capacità di informare i team responsabili del carico di lavoro quando un pacchetto deve essere aggiornato in base alle informazioni su chi usa cosa.
- Minor rischio di includere nel software un pacchetto con problemi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Fornisci servizi centralizzati per i pacchetti e le dipendenze in modo da semplificarne l'uso per gli sviluppatori. La centralizzazione dei servizi può essere eseguita in modo logico anziché implementarli come sistema monolitico. Questo approccio permette di fornire servizi in modo da soddisfare le esigenze degli sviluppatori. Devi implementare una soluzione ottimale per l'aggiunta di pacchetti al repository in caso di aggiornamenti o nuovi requisiti. Servizi AWS come AWS CodeArtifact o soluzioni simili dei partner AWS forniscono tale funzionalità.

Passaggi dell'implementazione

- Implementa un servizio di repository centralizzato in modo logico che sia disponibile in tutti gli ambienti in cui viene sviluppato il software.
- Includi l'accesso al repository come parte del processo di provisioning automatico dell'Account AWS.
- Crea automazione per testare i pacchetti prima della loro pubblicazione in un repository.
- Gestisci le metriche dei pacchetti, dei linguaggi e dei team usati più comunemente e con la maggiore quantità di modifiche.
- Offri ai team di sviluppo un meccanismo automatico per richiedere nuovi pacchetti e fornire feedback.
- Analizza regolarmente i pacchetti nel repository per identificare il possibile impatto di nuovi problemi riscontrati.

Risorse

Best practice correlate:

- SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test

Documenti correlati:

- DevOps Guidance: DL.CS.2 Sign code artifacts after each build
- Supply chain Levels for Software Artifacts (SLSA)

Esempi correlati:

- Accelerate deployments on AWS with effective governance
- Tighten your package security with CodeArtifact Package Origin Control toolkit
- Multi Region Package Publishing Pipeline (GitHub)
- Publishing Node.js Modules on AWS CodeArtifact using AWS CodePipeline (GitHub)
- AWS CDK Java CodeArtifact Pipeline Sample (GitHub)
- Distribute private .NET NuGet packages with AWS CodeArtifact (GitHub)

Video correlati:

- Proactive security: Considerations and approaches
- The AWS Philosophy of Security (re:Invent 2017)
- When security, safety, and urgency all matter: Handling Log4Shell

SEC11-BP06 Implementazione programmatica del software

Esegui implementazioni programmatiche del software laddove possibile. Questo approccio riduce la probabilità che un'implementazione non riesca o che si verifichi un problema imprevisto a causa dell'errore umano.

Risultato desiderato: la versione del carico di lavoro da testare è la stessa che viene implementata e l'implementazione viene eseguita in modo coerente ogni volta. L'esternalizzazione della configurazione del carico di lavoro è utile per eseguirne l'implementazione in ambienti diversi senza modifiche. Viene utilizzata la firma crittografica dei pacchetti software per verificare che non vi siano cambiamenti da un ambiente all'altro.

Anti-pattern comuni:

- Implementazione manuale del software nell'ambiente di produzione.
- Applicazione manuale di modifiche al software per soddisfare i requisiti di ambienti diversi.

Vantaggi dell'adozione di questa best practice:

- Maggiore affidabilità del processo di rilascio del software.
- Riduzione dei rischi legati a modifiche errate che hanno impatto sulla funzionalità aziendale.
- Processi di rilascio più frequenti grazie a un rischio di modifica minimo.
- Funzionalità di rollback automatiche in caso di eventi imprevisti durante l'implementazione.
- Possibilità di usare la crittografia per dimostrare che il software implementato è esattamente identico a quello testato.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per mantenere un'infrastruttura applicativa solida e affidabile, implementa pratiche per l'implementazione sicura e automatizzata. Tali pratiche prevedono la rimozione dell'accesso umano persistente dagli ambienti di produzione, l'utilizzo di strumenti CI/CD per le implementazioni e l'esternalizzazione dei dati di configurazione specifici dell'ambiente. Seguendo questo approccio, è possibile migliorare la sicurezza, ridurre il rischio di errori umani e semplificare il processo di implementazione.

È possibile creare una struttura di Account AWS per rimuovere l'accesso umano persistente dagli ambienti di produzione. Questa pratica riduce al minimo il rischio di modifiche non autorizzate o accidentali, migliorando l'integrità dei sistemi di produzione. Invece dell'accesso umano diretto, puoi utilizzare strumenti CI/CD come AWS CodeBuild e AWS CodePipeline per eseguire le implementazioni. È possibile utilizzare questi servizi per automatizzare i processi di sviluppo, test e implementazione, riducendo l'intervento manuale e aumentando la coerenza.

Per migliorare ulteriormente la sicurezza e la tracciabilità, puoi firmare i pacchetti applicativi dopo che sono stati testati e convalidare le firme durante l'implementazione. A tale scopo, puoi usare strumenti crittografici come AWS Signer o AWS Key Management Service (AWS KMS). Firmando e verificando i pacchetti, puoi assicurarti di distribuire solo codice autorizzato e convalidato nei tuoi ambienti.

Inoltre, il tuo team può progettare il carico di lavoro per ottenere dati di configurazione specifici dell'ambiente da una fonte esterna, come AWS Systems Manager Parameter Store. Questa pratica separa il codice dell'applicazione dai dati di configurazione, il che consente di gestire e aggiornare le configurazioni in modo indipendente senza modificare il codice applicativo stesso.

Per semplificare il provisioning e la gestione dell'infrastruttura, valuta la possibilità di utilizzare strumenti di infrastructure as code (IaC) come AWS CloudFormation o AWS CDK. Puoi utilizzare

questi strumenti per definire l'infrastruttura come codice, con conseguente miglioramento della coerenza e della ripetibilità delle implementazioni in ambienti diversi.

Prendi in considerazione le distribuzioni canary per convalidare la corretta implementazione del tuo software. Le distribuzioni canary prevedono l'implementazione delle modifiche in un sottoinsieme di istanze o utenti prima dell'implementazione nell'intero ambiente di produzione. È quindi possibile monitorare l'impatto delle modifiche ed eventualmente annullarle, se necessario, in modo da ridurre al minimo il rischio di problemi diffusi.

Segui i consigli delineati nel white paper Organization Your AWS Environment Using Multiple Accounts. Questo white paper fornisce indicazioni su come suddividere gli ambienti (ad esempio, tra ambiente di sviluppo, di gestione temporanea e di produzione) in Account AWS distinti, con conseguente ulteriore miglioramento della sicurezza e dell'isolamento.

Passaggi dell'implementazione

1. Configurazione della struttura di Account AWS:
   - Segui le indicazioni contenute nel white paper Organization Your AWS Environment Using Multiple Accounts per creare Account AWS separati per ambienti diversi (ad esempio, ambiente di sviluppo, di gestione temporanea e di produzione).
   - Configura le autorizzazioni e i controlli di accesso appropriati per ogni account per limitare l'accesso umano diretto agli ambienti di produzione.
2. Implementa una pipeline CI/CD:
   - Configura una pipeline CI/CD utilizzando servizi come AWS CodeBuild e AWS CodePipeline.
   - Configura la pipeline per creare, testare e implementare automaticamente il codice applicativo nei rispettivi ambienti.
   - Integra i repository di codice con la pipeline CI/CD per il controllo delle versioni e la gestione del codice.
3. Firma e verifica i pacchetti applicativi:
   - Usa AWS Signer o AWS Key Management Service (AWS KMS) per firmare i pacchetti applicativi dopo che sono stati testati e convalidati.
   - Configura il processo di implementazione per verificare le firme dei pacchetti applicativi prima di distribuirli negli ambienti di destinazione.
4. Esternalizza i dati di configurazione:
   - Archivia i dati di configurazione specifici dell'ambiente in AWS Systems Manager Parameter Store.

- Modifica il codice applicativo per recuperare i dati di configurazione dal Parameter Store durante l'implementazione o il runtime.

5. Implementa l'infrastructure as code (IaC):
   - Usa strumenti IaC come AWS CloudFormation o AWS CDK per definire e gestire la tua infrastruttura come codice.
   - Crea modelli CloudFormation o script CDK per fornire e configurare le risorse AWS necessarie per la tua applicazione.
   - Integra l'IaC con la tua pipeline CI/CD per implementare automaticamente le modifiche all'infrastruttura insieme alle modifiche al codice applicativo.
6. Implementa la distribuzione canary:
   - Configura il processo di implementazione per supportare le distribuzioni canary, in cui le modifiche vengono implementate in un sottoinsieme di istanze o utenti prima dell'implementazione nell'intero ambiente di produzione.
   - Utilizza servizi come AWS CodeDeploy o AWS ECS per gestire le distribuzioni canary e monitorare l'impatto delle modifiche.
   - Implementa meccanismi di rollback per tornare alla precedente versione stabile qualora vengano rilevati problemi durante la distribuzione canary.
7. Monitora ed esegui audit:
   - Configura meccanismi di monitoraggio e registrazione di log per tenere traccia delle implementazioni, delle prestazioni delle applicazioni e delle modifiche all'infrastruttura.
   - Usa servizi come Amazon CloudWatch e AWS CloudTrail per raccogliere e analizzare log e metriche.
   - Implementa controlli di conformità e audit per verificare l'aderenza alle best practice e ai requisiti normativi in materia di sicurezza.
8. Migliora continuamente i processi:
   - Rivedi e aggiorna regolarmente le tue pratiche di implementazione, integrando feedback e informazioni apprese dalle implementazioni precedenti.
   - Automatizza il più possibile il processo di implementazione per ridurre l'intervento manuale e i potenziali errori umani.
   - Collabora con team interfunzionali (ad esempio, operativi o di sicurezza) per allineare e migliorare continuamente le pratiche di implementazione.

Seguendo questi passaggi, puoi mettere in atto pratiche di implementazione sicure e automatizzate nel tuo ambiente AWS, migliorando la sicurezza, riducendo il rischio di errori umani e semplificando il processo di implementazione.

Risorse

Best practice correlate:

- SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test
- DL.CI.2 Trigger builds automatically upon source code modifications

Documenti correlati:

- Accelerate deployments on AWS with effective governance
- Automatizzazione di distribuzioni pratiche e sicure
- Code signing using AWS Certificate Manager Private CA and AWS Key Management Service asymmetric keys
- Code Signing, a Trust and Integrity Control for AWS Lambda

Video correlati:

- Hands-off: Automating continuous delivery pipelines at Amazon

Esempi correlati:

- Blue/Green deployments with AWS Fargate

SEC11-BP07 Valutazione regolare delle proprietà di sicurezza delle pipeline

Applica i principi del pilastro della sicurezza Well-Architected alle pipeline, con particolare attenzione alla separazione delle autorizzazioni. Valuta regolarmente le proprietà di sicurezza della tua infrastruttura di pipeline. Una gestione efficace della sicurezza delle pipeline assicura la protezione del software che passa attraverso le pipeline.

Risultato desiderato: le pipeline in uso per la creazione e implementazione del tuo software seguono le stesse pratiche consigliate applicate per qualsiasi altro carico di lavoro nel tuo ambiente. I test che vengono implementati nelle pipeline non sono modificabili dai team che li utilizzano. Alle pipeline vengono assegnate solo le autorizzazioni necessarie per le implementazioni in esecuzione

utilizzando credenziali temporanee. Vengono implementate misure di sicurezza per impedire che le pipeline vengano implementate negli ambienti sbagliati. Le pipeline vengono configurate in modo da comunicare lo stato, così da consentire la convalida dell'integrità degli ambienti di sviluppo.

Anti-pattern comuni:

- Test di sicurezza ignorabili dagli sviluppatori.
- Autorizzazioni eccessivamente elevate per le pipeline di implementazione.
- Pipeline non configurate per la convalida degli input.
- Nessuna revisione periodica delle autorizzazioni associate all'infrastruttura CI/CD.
- Uso di credenziali a lungo termine o hardcoded.

Vantaggi dell'adozione di questa best practice:

- Maggiore garanzia di integrità del software sviluppato e implementato attraverso le pipeline.
- Possibilità di arrestare un'implementazione in caso di attività sospetta.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Le pipeline di implementazione sono una componente fondamentale del ciclo di vita dello sviluppo del software e devono seguire gli stessi principi e le stesse pratiche di sicurezza di qualsiasi altro carico di lavoro nel tuo ambiente. Ciò include l'implementazione di controlli di accesso adeguati, la convalida degli input, oltre alla revisione e all'audit periodici delle autorizzazioni associate all'infrastruttura CI/CD.

Verifica che i team responsabili della creazione e della distribuzione delle applicazioni non siano in grado di modificare o aggirare i test e i controlli di sicurezza implementati nelle pipeline. Questa separazione delle responsabilità aiuta a mantenere l'integrità dei processi di creazione e implementazione.

Come punto di partenza, valuta la possibilità di utilizzare l'architettura AWS di riferimento per le pipeline di implementazione. Questa architettura di riferimento offre una base sicura e scalabile per la creazione di pipeline CI/CD su AWS.

Inoltre, è possibile utilizzare servizi come AWS Identity and Access Management Access Analyzer per generare policy IAM con privilegio minimo sia per le autorizzazioni delle pipeline sia per

l'esecuzione di una fase delle pipeline destinata a verificare le autorizzazioni dei carichi di lavoro. Tutto questo consente di verificare che le pipeline e i carichi di lavoro dispongano solo delle autorizzazioni necessarie per le rispettive funzioni specifiche, riducendo il rischio di azioni o accessi non autorizzati.

Passaggi dell'implementazione

- Parti dall'architettura di riferimento per le pipeline di implementazione AWS.
- Prendi in considerazione l'utilizzo di AWS IAM Access Analyzer per generare in modo programmatico policy IAM con privilegio minimo per le pipeline.
- Integra nelle tue pipeline monitoraggio e avvisi in modo da ricevere notifiche in caso di attività impreviste o anomale, per i servizi AWS gestiti. Amazon EventBridge ti consente di indirizzare i dati verso destinazioni come AWS Lambda o Amazon Simple Notification Service (Amazon SNS).

Risorse

Documenti correlati:

- AWS Architettura di riferimento per pipeline di implementazione
- Monitoraggio di AWS CodePipeline
- Best practice di sicurezza per AWS CodePipeline

Esempi correlati:

- DevOps monitoring dashboard (GitHub)

SEC11-BP08 Creazione di un programma per l'integrazione della titolarità della sicurezza nei team responsabili del carico di lavoro

Crea un programma o un meccanismo che permetta ai team di sviluppo di prendere decisioni sulla sicurezza del software che creano. Il team della sicurezza dovrà convalidare queste decisioni durante una revisione, ma integrare la proprietà della sicurezza nei team di sviluppo consente di creare carichi di lavoro più veloci e sicuri. Questo meccanismo promuove anche una cultura della responsabilità che ha un impatto positivo sul funzionamento dei sistemi che crei.

Risultato desiderato: integrazione della titolarità della sicurezza e dei processi decisionali correlati nei team. I tuoi team hanno ricevuto una formazione sul corretto approccio alla sicurezza oppure sono

stati ampliati con personale addetto alla sicurezza integrato o associato. Di conseguenza, i team prendono decisioni migliori sulla sicurezza nelle fasi iniziali del ciclo di sviluppo.

Anti-pattern comuni:

- Assegnazione di tutte le decisioni in materia di sicurezza al team responsabile della sicurezza.
- Gestione dei requisiti di sicurezza in fasi tardive del processo di sviluppo.
- Assenza di feedback di sviluppatori e responsabili della sicurezza sul funzionamento del programma.

Vantaggi dell'adozione di questa best practice:

- Riduzione del tempo necessario per completare le revisioni della sicurezza.
- Riduzione dei problemi di sicurezza rilevati solo in fase di revisione della sicurezza.
- Miglioramento della qualità complessiva del software compilato.
- Opportunità di identificare e comprendere i problemi sistematici o le aree di miglioramento a valore elevato.
- Riduzione della quantità di attività di correzione dovute agli esiti delle revisioni della sicurezza.
- Migliore percezione della funzione della sicurezza.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Per iniziare, attieniti alle linee guida illustrate in SEC11-BP01 Formazione per la sicurezza delle applicazioni Identifica quindi il modello operativo per il programma che ritieni più efficace per l'organizzazione. I due modelli principali consistono nel formare gli sviluppatori o nell'integrare responsabili della sicurezza nei team di sviluppo. Una volta scelto l'approccio iniziale, devi eseguire un progetto pilota con un singolo team o un piccolo gruppo di team del carico di lavoro per dimostrare il funzionamento del modello per l'organizzazione. Il supporto autorevole da parte dello sviluppatore e di altre parti responsabili della sicurezza dell'organizzazione semplifica l'implementazione e il successo del programma. Durante la creazione del programma, è importante scegliere le metriche da usare per dimostrarne il valore. Per un'ottima esperienza formativa, puoi documentarti sul modo in cui AWS ha affrontato questo problema. Questa best practice è per lo più incentrata sulla trasformazione e sulla cultura aziendali. Gli strumenti usati devono supportare la collaborazione tra lo sviluppatore e le comunità responsabili della sicurezza.

Passaggi dell'implementazione

- Per iniziare, predisponi corsi di formazione sulla sicurezza delle applicazioni per gli sviluppatori.
- Crea una community e un programma di onboarding per formare gli sviluppatori.
- Scegli un nome per il programma. Alcuni termini comunemente usati sono Responsabilità, Supporto o Promozione.
- Identifica il modello da usare: formazione per gli sviluppatori, integrazione di tecnici della sicurezza o ruoli di sicurezza per affinità.
- Identifica alcuni sponsor del progetto tra responsabili della sicurezza, sviluppatori e altri gruppi potenzialmente pertinenti.
- Tieni traccia delle metriche per il numero di persone coinvolte nel programma, del tempo impiegato per le revisioni e del feedback ottenuto da sviluppatori e responsabili della sicurezza. Usa queste metriche per apportare miglioramenti.

Risorse

Best practice correlate:

- SEC11-BP01 Formazione per la sicurezza delle applicazioni
- SEC11-BP02 Automazione dei test lungo il ciclo di vita di sviluppo e test

Documenti correlati:

- How to approach threat modeling
- How to think about cloud security governance
- How AWS built the Security Guardians program, a mechanism to distribute security ownership
- How to build a Security Guardians program to distribute security ownership

Video correlati:

- Proactive security: Considerations and approaches
- AppSec tooling and culture tips from AWS and Toyota Motor North America

Affidabilità

Il pilastro dell'affidabilità comprende la capacità di un carico di lavoro di eseguire la funzione attesa in modo corretto e coerente quando previsto. Puoi trovare linee guida prescrittive sull'implementazione nel whitepaper sul pilastro dell'affidabilità.

Aree delle best practice

- Fondamenti
- Architettura del carico di lavoro
- Gestione delle modifiche
- Gestione dei guasti

Fondamenti

Questions

- REL 1. Come si gestiscono Service Quotas e restrizioni?
- REL 2. Come si pianifica la topologia della rete?

REL 1. Come si gestiscono Service Quotas e restrizioni?

Per le architetture di carichi di lavoro basate sul cloud, esistono Service Quotas (chiamate anche restrizioni dei servizi). Queste quote sono presenti per evitare di effettuare accidentalmente il provisioning di più risorse di quelle necessarie e limitare i tassi di richiesta sulle operazioni API in modo da proteggere i servizi da un uso illecito. Esistono anche vincoli di risorse, ad esempio la velocità con cui è possibile trasferire i bit su un cavo in fibra ottica o lo spazio di archiviazione su un disco fisico.

Best practice

- REL01-BP01 Consapevolezza su quote e vincoli di servizio
- REL01-BP02 Gestione delle quote di servizio in più account e regioni
- REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura
- REL01-BP04 Monitoraggio e gestione delle quote
- REL01-BP05 Automazione della gestione delle quote

- REL01-BP06 Creazione di un divario sufficiente tra le quote attuali e l'utilizzo massimo per consentire eventuali failover

REL01-BP01 Consapevolezza su quote e vincoli di servizio

Conosci le quote predefinite e gestisci le richieste di aumento delle quote per l'architettura del carico di lavoro. Sai quali vincoli delle risorse cloud, ad esempio disco o rete, sono potenzialmente influenti.

Risultato desiderato: i clienti possono prevenire il degrado o l'interruzione nei loro Account AWS mediante l'implementazione di linee guida opportune per il monitoraggio delle metriche chiave, gli esami dell'infrastruttura e le misure di correzione dell'automazione, così da verificare che non vengano raggiunte quote e vincoli di servizio che potrebbero causare degrado o interruzione del servizio.

Anti-pattern comuni:

- Distribuzione di un carico di lavoro senza comprendere le quote hard o soft e i relativi limiti per i servizi utilizzati.
- Distribuzione di un carico di lavoro sostitutivo senza analizzare e riconfigurare le quote necessarie o contattare preventivamente l'assistenza.
- Supposizione che i servizi cloud non abbiano limiti e che i servizi possano essere utilizzati senza tener conto di tariffe, limiti, conteggi, quantità.
- Supposizione che le quote verranno aumentate automaticamente.
- Mancata conoscenza del processo e della scadenza delle richieste di quote.
- Supposizione che la quota predefinita del servizio cloud sia identica per ogni servizio rispetto alle varie regioni.
- Supposizione che i vincoli del servizio possano essere violati e che i sistemi procedano al dimensionamento automatico o aumentino il limite oltre i vincoli della risorsa
- Nessun test dell'applicazione nei momenti di picco del traffico, per stressare l'utilizzo delle sue risorse.
- Provisioning della risorsa senza analisi della dimensione della risorsa richiesta.
- Provisioning in eccesso di capacità scegliendo tipi di risorse che vanno ben oltre il fabbisogno effettivo o i picchi previsti.
- Nessuna valutazione dei requisiti di capacità per nuovi livelli di traffico prima di un nuovo evento cliente o dell'implementazione di una nuova tecnologia.

Vantaggi dell'adozione di questa best practice: il monitoraggio e la gestione automatizzata di quote di servizio e vincoli di risorse consentono di ridurre in modo proattivo i guasti. Le modifiche nei modelli di traffico per il servizio di un cliente possono causare un'interruzione o un degrado se non si seguono le best practice. Monitorando e gestendo questi valori in tutte le regioni e in tutti gli account, le applicazioni possono avere una maggiore resilienza in caso di eventi avversi o non pianificati.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Service Quotas è un servizio AWS che consente di gestire le quote per 250 servizi AWS da un'unica posizione. Oltre a cercare i valori delle quote, si possono anche richiedere e monitorare gli aumenti delle stesse tramite la console di Service Quotas o l'SDK AWS. AWS Trusted Advisor offre un controllo delle quote di servizio che mostra l'utilizzo e le quote per alcuni aspetti di determinati servizi. Le quote di servizio predefinite per servizio sono indicate anche nella documentazione AWS di ciascun servizio (consulta, ad esempio, le quote di Amazon VPC).

Alcuni limiti dei servizi, come i limiti di velocità sulle API con limitazione (della larghezza di banda della rete) vengono impostati all'interno del Gateway Amazon API stesso configurando un piano di utilizzo. Altre restrizioni impostate come configurazione per i rispettivi servizi includono capacità di IOPS allocata, storage Amazon RDS allocato e allocazioni di volumi EBS. Amazon Elastic Compute Cloud dispone di un proprio pannello di controllo sulle restrizioni dei servizi che consente di gestire l'istanza, Amazon Elastic Block Store e i limiti degli indirizzi IP elastici. Se hai un caso d'uso in cui le quote di servizio influiscono sulle prestazioni della tua applicazione e queste non sono adattabili alle tue esigenze, contatta Supporto per scoprire se sono possibili riduzioni.

Le quote di servizio possono essere specifiche per ogni regione o di natura globale. L'uso di un servizio AWS che raggiunge la sua quota non si comporterà come previsto nell'uso normale e potrebbe causare interruzioni o degrado del servizio. Ad esempio, una quota di servizio stabilisce limiti per il numero di istanze DL Amazon EC2 impiegate in una regione. È possibile raggiungere tale limite durante un evento di scalabilità del traffico utilizzando i gruppi Auto Scaling (ASG).

Le quote di servizio per ogni account devono essere valutate in modo regolare per determinare i limiti di servizio opportuni per quell'account. Queste quote di servizio fungono da guardrail operativi, per evitare di fornire accidentalmente più risorse di quelle necessarie. Servono anche a limitare i tassi di richiesta delle operazioni API per proteggere i servizi dagli abusi.

I limiti dei servizi sono diversi dalle quote dei servizi. I vincoli di servizio rappresentano i limiti di una particolare risorsa, definiti dalla stessa. Questi possono essere la capacità di archiviazione (ad

esempio, gp2 ha un limite di dimensione di 1 GB - 16 TB) o il throughput del disco. È essenziale che il vincolo di un tipo di risorsa sia progettato e valutato in modo costante per l'utilizzo che potrebbe raggiungere il suo limite. In caso di raggiungimento inaspettato di un vincolo, può verificarsi il degrado o l'interruzione delle applicazioni o dei servizi dell'account.

Se uno dei tuoi casi d'uso presenta quote di servizio che influiscono sulle prestazioni della tua applicazione e non sono adattabili alle tue esigenze, contatta Supporto per vedere se sono possibili mitigazioni. Per ulteriori dettagli sull'adeguamento delle quote fisse, consulta REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura.

Esistono alcuni servizi e strumenti AWS per il monitoraggio e la gestione di Service Quotas. Sfrutta il servizio e gli strumenti per fornire controlli automatizzati o manuali dei livelli di quota.

- AWS Trusted Advisor offre un controllo delle quote di servizio che mostra l'utilizzo e le quote per alcuni aspetti di determinati servizi. Può aiutare a identificare i servizi vicini alle quote.
- Console di gestione AWS fornisce metodi per la visualizzazione dei valori delle quote dei servizi, la gestione, la richiesta di nuove quote, il monitoraggio dello stato delle richieste di quote e la visualizzazione della cronologia delle quote.
- AWS CLI e CDK offrono metodi programmatici per gestire e monitorare in automatico utilizzo e livelli delle quote di servizio.

Passaggi dell'implementazione

Per Service Quotas:

- Consulta AWS Service Quotas.
- Per avere la certezza delle quote di servizio esistenti, stabilisci i servizi (come IAM Access Analyzer) usati. Sono circa 250 i servizi AWS controllati da quote di servizio. Quindi stabilisci il nome della quota di servizio specifica utilizzabile all'interno di ogni account e regione. Esistono circa 3000 nomi di quote di servizio per regione.
- Amplia questa analisi delle quote con AWS Config per individuare tutte le risorse AWS utilizzate nei tuoi Account AWS.
- Usa i dati di AWS CloudFormation per determinare le risorse AWS utilizzate. Esamina le risorse create in Console di gestione AWS o con il comando AWS CLI di `list-stack-resources`. È anche possibile vedere le risorse configurate da implementare nel modello stesso.
- Stabilisci tutti i servizi necessari per il tuo carico di lavoro analizzando il codice di implementazione.

- Determina le quote di servizio applicabili. Utilizza le informazioni accessibili in modo programmatico da Trusted Advisor e Service Quotas.
- Stabilisci un metodo di monitoraggio automatizzato (consulta REL01-BP02 Gestione delle quote di servizio in più account e regioni e REL01-BP04 Monitoraggio e gestione delle quote) per ricevere avvisi e informazioni se le quote di servizio sono vicine al limite o lo hanno superato.
- Stabilisci un metodo automatizzato e programmatico per verificare se una quota di servizio ha subito modifiche in una regione ma non in altre nello stesso account (consulta REL01-BP02 Gestione delle quote di servizio in più account e regioni e REL01-BP04 Monitoraggio e gestione delle quote).
- Automatizza la scansione dei log e delle metriche delle applicazioni per determinare la presenza di errori di quota o di vincoli di servizio. In presenza di errori, invia gli allarmi al sistema di monitoraggio.
- Stabilisci procedure di progettazione per calcolare la modifica richiesta della quota (consulta REL01-BP05 Automazione della gestione delle quote) una volta individuata la necessità di quote più elevate per servizi specifici.
- Crea un flusso di lavoro di provisioning e di approvazione per richiedere modifiche alla quota di servizio, che dovrebbe includere un flusso di lavoro di eccezione in caso di rifiuto della richiesta o di approvazione parziale.
- Crea un metodo di progettazione per rivedere le quote di servizio prima del provisioning e dell'utilizzo di nuovi servizi AWS prima del roll-out in ambienti di produzione o carichi (ad esempio, account di test di carico).

Per i vincoli dei servizi:

- Stabilisci metodi di monitoraggio e metriche per avvisi in caso di avvicinamento da parte delle risorse ai relativi limiti. Sfrutta CloudWatch in base alle necessità per le metriche o il monitoraggio dei log.
- Stabilisci soglie di allarme per ciascuna risorsa con un vincolo significativo per l'applicazione o il sistema.
- Crea procedure di gestione del flusso di lavoro e dell'infrastruttura per cambiare il tipo di risorsa se il vincolo è prossimo all'utilizzo. Questo flusso di lavoro dovrebbe includere test di carico come best practice per verificare che quello nuovo sia il tipo di risorsa corretto in base ai nuovi vincoli.
- Migra la risorsa identificata al nuovo tipo di risorsa consigliato, utilizzando procedure e processi esistenti.

Risorse

Best practice correlate:

- REL01-BP02 Gestione delle quote di servizio in più account e regioni
- REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura
- REL01-BP04 Monitoraggio e gestione delle quote
- REL01-BP05 Automazione della gestione delle quote
- REL01-BP06 Creazione di un divario sufficiente tra le quote attuali e l'utilizzo massimo per consentire eventuali failover
- REL03-BP01 Scegli come segmentare il tuo carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL12-BP04 Test della resilienza tramite l’utilizzo dell’ingegneria del caos

Documenti correlati:

- AWS Pilastro dell’affidabilità di Well-Architected Framework: disponibilità
- AWS Service Quotas (precedentemente definite restrizioni dei servizi)
- AWS Trusted Advisor Best Practice Checks (consulta la sezione Service Limits)
- AWS limit monitor on AWS answers
- Amazon EC2 Service Limits
- What is Service Quotas?
- How to Request Quota Increase
- Service endpoints and quotas
- Guida per l’utente di Service Quotas
- Quota Monitor for AWS
- Limiti di isolamento dei guasti di AWS
- Availability with redundancy
- AWS for Data
- Cos’è l’integrazione continua?
- Cos’è la distribuzione continua?

- Partner APN: partner per la gestione della configurazione
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS
- Managing and monitoring API throttling in your workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower

Video correlati:

- AWS Live re:Inforce 2019 - Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas
- AWS IAM Quotas Demo

Strumenti correlati:

- Revisore Amazon CodeGuru
- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- Marketplace AWS

REL01-BP02 Gestione delle quote di servizio in più account e regioni

Se utilizzi più account o regioni, assicurati di richiedere le quote opportune in tutti gli ambienti di esecuzione dei carichi di lavoro di produzione.

Risultato desiderato: servizi e applicazioni non dovrebbero essere influenzati dall'esaurimento della quota di servizio per le configurazioni che si estendono su account o regioni o che presentano progetti di resilienza che utilizzano il failover di zona, regione o account.

Anti-pattern comuni:

- Si consente l'aumento dell'utilizzo delle risorse in una regione di isolamento senza alcun meccanismo per mantenere la capacità nelle altre.
- Impostazione manualmente tutte le quote in modo indipendente nelle regioni di isolamento.
- Mancata valutazione dell'effetto delle architetture di resilienza (come quelle attive o passive) nelle future esigenze di quote durante un degrado nella regione non primaria.
- Mancata valutazione regolare delle quote e applicazione dele modifiche necessarie in ogni regione e account in cui viene gestito il carico di lavoro.
- Mancato utilizzo dei modelli di richiesta di quote per la richiesta di aumenti in più regioni e account.
- Mancato aggiornamento delle quote dei servizi, perché si pensa erroneamente che l'aumento delle quote abbia implicazioni di costo, come le richieste di prenotazione di calcolo.

Vantaggi dell'adozione di questa best practice: verifica della capacità di gestire il carico corrente nelle regioni o negli account secondari in caso di indisponibilità dei servizi regionali. Questo consente di ridurre il numero di errori o livelli di degrado che si verificano durante la perdita di regioni.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Il monitoraggio delle quote di servizio avviene per account. Salvo diversa indicazione, ogni quota è specifica della Regione AWS. Oltre agli ambienti di produzione, gestisci anche le quote in tutti gli ambienti non di produzione applicabili, in modo che test e sviluppo non siano ostacolati. Il mantenimento di un elevato grado di resilienza richiede una valutazione continua delle quote di servizio (sia automatica che manuale).

Con un aumento dei carichi di lavoro in tutte le regioni dovuto all'implementazione di progetti che utilizzano approcci attivo/attivo, attivo/passivo con standby a caldo, attivo/passivo con standby a freddo e attivo/passivo con Pilot Light, è essenziale conoscere tutti i livelli di quota di regione e account. I modelli di traffico passati non sono sempre un buon indicatore per stabilire se la quota di servizio è impostata correttamente.

Altrettanto importante è il fatto che il limite di nome della quota di servizio non è sempre lo stesso per ogni regione. In una regione, il valore potrebbe essere cinque, in un'altra potrebbe essere dieci. La gestione di queste quote deve riguardare tutti gli stessi servizi, account e regioni per garantire una resilienza costante sotto carico.

Riconcilia tutte le differenze di quota di servizio tra le diverse regioni (regione attiva o passiva) e crea processi per riconciliare continuamente queste differenze. I piani di test dei failover passivi delle regioni sono raramente scalati in base alla capacità attiva di picco, il che significa che gli esercizi delle giornate di gioco o table top potrebbero non riuscire a trovare le differenze nelle quote di servizio tra le regioni e a mantenere i limiti corretti.

La deviazione della quota di servizio, la condizione in cui la modifica dei limiti della quota di servizio per una determinata quota denominata avviene in una regione e non in tutte le regioni, è un fattore molto importante da monitorare e valutare. Si dovrebbe prendere in considerazione la possibilità di modificare la quota nelle regioni con traffico o potenzialmente in grado di trasportare traffico.

- Seleziona account e regioni pertinenti in base ai tuoi requisiti di servizio, latenza, normativi e disaster recovery.
- Identifica le quote dei servizi per tutti gli account, le regioni e le zone di disponibilità pertinenti. Le restrizioni si riferiscono ad account e regione. Confronta questi valori per individuare le differenze.

Passaggi dell'implementazione

- Rivedi i valori di Service Quotas che potrebbero aver superato il livello di rischio di utilizzo. AWS Trusted Advisor offre allarmi per la violazione di soglie dell'80% e del 90%.
- Rivedi i valori per le quote di servizio in qualsiasi regione passiva (in un progetto Attivo/Passivo). Verifica che il carico venga eseguito in modo corretto nelle regioni secondarie in caso di guasto nella regione primaria.
- Valuta in modo automatizzato se si è verificata una deviazione delle quote di servizio tra le regioni dello stesso account e agisci di conseguenza per modificare i limiti.
- Se le unità organizzative (UO) del cliente sono strutturate nel modo supportato, aggiorna i modelli di quote di servizio per riflettere le modifiche alle quote da applicare a più regioni e account.
  - Crea un modello e associa le regioni alla modifica della quota.
  - Rivedi tutti i modelli delle quote di servizio esistenti per qualsiasi modifica richiesta (regione, limiti e account).

Risorse

Best practice correlate:

- REL01-BP01 Consapevolezza su quote e vincoli di servizio

- REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura
- REL01-BP04 Monitoraggio e gestione delle quote
- REL01-BP05 Automazione della gestione delle quote
- REL01-BP06 Creazione di un divario sufficiente tra le quote attuali e l'utilizzo massimo per consentire eventuali failover
- REL03-BP01 Scegli come segmentare il tuo carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL12-BP04 Test della resilienza tramite l'utilizzo dell'ingegneria del caos

Documenti correlati:

- Pilastro dell'affidabilità di AWS Well-Architected Framework: disponibilità
- AWS Service Quotas (precedentemente definite restrizioni dei servizi)
- AWS Trusted Advisor Best Practice Checks (consulta la sezione Service Limits)
- AWS limit monitor on AWS answers
- Amazon EC2 Service Limits
- What is Service Quotas?
- How to Request Quota Increase
- Service endpoints and quotas
- Guida per l'utente di Service Quotas
- Quota Monitor for AWS
- Limiti di isolamento dei guasti di AWS
- Availability with redundancy
- AWS for Data
- Cos'è l'integrazione continua?
- Cos'è la distribuzione continua?
- Partner APN: partner per la gestione della configurazione
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS

- Managing and monitoring API throttling in your workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower

Video correlati:

- AWS Live re:Inforce 2019 - Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas
- AWS IAM Quotas Demo

Servizi correlati:

- Revisore Amazon CodeGuru
- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- Marketplace AWS

REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura

Identifica con attenzione quote di servizio, vincoli del servizio e limiti delle risorse fisiche che non possono essere modificati. Progetta architetture per applicazioni e servizi in modo da impedire che questi limiti pregiudichino l'affidabilità.

Alcuni esempi includono la larghezza di banda della rete, le dimensioni di payload dell'invocazione di funzioni serverless, il tasso di espansione della limitazione (della larghezza di banda della rete) per un gateway API e le connessioni utente simultanee a un database.

Risultato desiderato: funzionamento dell'applicazione o del servizio come previsto in condizioni di traffico normale e intenso. L'applicazione o il servizio è stato progettato per operare entro i limiti dei vincoli o delle quote di servizio fissi della risorsa.

Anti-pattern comuni:

- Scelta di una progettazione che usa una risorsa di un servizio, senza essere al corrente della presenza di vincoli che causeranno errori di progettazione durante il dimensionamento.
- Esecuzione di benchmark poco realistici e che raggiungono le quote di servizio fisse durante i test. Ad esempio, l'esecuzione di test a un limite di espansione per un periodo di tempo prolungato.
- Scelta di una progettazione non scalabile o modificabile in caso di superamento delle quote di servizio fisse. Ad esempio, dimensioni dei payload SQS di 256 KB.
- Mancata progettazione e implementazione dell'osservabilità per monitorare e inviare avvisi circa le soglie per le quote di servizio a rischio durante eventi di traffico elevato.

Vantaggi dell'adozione di questa best practice: verifica del funzionamento dell'applicazione con tutti i livelli di carico dei servizi previsti senza interruzioni o deterioramenti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Diversamente dalle risorse e dalle quote di servizio flessibili, sostituibili con unità di capacità maggiori, le quote di servizio fisse in AWS non possono essere modificate. Di conseguenza, occorre verificare tutti i servizi AWS di questo tipo per identificare i possibili limiti fissi di capacità in caso di relativo utilizzo per la progettazione di un'applicazione.

I limiti fissi vengono mostrati nella console di Service Quotas. Se le colonne visualizzano `ADJUSTABLE = No`, il servizio ha un limite fisso. I limiti fissi vengono mostrati anche in alcune pagine di configurazione delle risorse. Ad esempio, Lambda presenta un limite fisso specifico che non può essere modificato.

Ad esempio, durante la progettazione di un'applicazione Python da eseguire in una funzione Lambda, l'applicazione deve essere valutata per determinare la probabilità di un'esecuzione di Lambda superiore a 15 minuti. Se il codice potrebbe restare in esecuzione oltre questo limite della quota di servizio, devi prendere in considerazione tecnologie o progettazioni alternative. In caso di raggiungimento del limite dopo l'implementazione nell'ambiente di produzione, l'applicazione sarà soggetta a errori o interruzioni fino alla correzione. A differenza dalle quote flessibili, non esiste alcun metodo per modificare i limiti, anche in caso di eventi di emergenza con livello di gravità 1.

Una volta implementata l'applicazione in un ambiente di test, occorre adottare una strategia per determinare se l'eventuale probabilità di raggiungere i limiti fissi. I test di stress, di carico e di caos devono fare parte del piano di test iniziale.

Passaggi dell'implementazione

- Esamina l'elenco completo dei servizi AWS utilizzabili nella fase di progettazione dell'applicazione.
- Esamina i limiti di quota flessibili e fissi per tutti i servizi. Non tutti i limiti vengono mostrati nella console di Service Quotas. Alcuni servizi indicano tali limiti in posizioni alternative.
- Nel progettare l'applicazione, esamina i principali fattori commerciali e tecnologici del carico di lavoro, come risultati aziendali, casi d'uso, sistemi dipendenti, obiettivi di disponibilità e oggetti di disaster recovery. Orienta il processo di identificazione del sistema distribuito corretto per il carico di lavoro in base a tali fattori commerciali e tecnologici.
- Analizza il carico dei servizi tra regioni e account. Molti limiti fissi per i servizi variano a seconda della regione. Tuttavia, alcuni limiti dipendono dagli account.
- Analizza le architetture di resilienza per l'utilizzo delle risorse durante un guasto a livello di zona e di regione. Nel corso dello sviluppo di progettazioni multi-regione che usano approcci attivo/attivo, attivo/passivo con standby a caldo, attivo/passivo con standby a freddo e attivo/passivo con Pilot Light, i casi di errore determineranno un utilizzo più elevato. Questo comportamento crea un possibile caso d'uso per il raggiungimento dei limiti fissi.

Risorse

Best practice correlate:

- REL01-BP01 Consapevolezza su quote e vincoli di servizio
- REL01-BP02 Gestione delle quote di servizio in più account e regioni
- REL01-BP04 Monitoraggio e gestione delle quote
- REL01-BP05 Automazione della gestione delle quote
- REL01-BP06 Creazione di un divario sufficiente tra le quote attuali e l'utilizzo massimo per consentire eventuali failover
- REL03-BP01 Scegli come segmentare il tuo carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli

- REL12-BP04 Test della resilienza tramite l'utilizzo dell'ingegneria del caos

Documenti correlati:

- AWS Pilastro dell'affidabilità di Well-Architected Framework: disponibilità
- AWS Service Quotas (precedentemente definite restrizioni dei servizi)
- AWS Trusted Advisor Best Practice Checks (consulta la sezione Service Limits)
- AWS limit monitor on AWS answers
- Amazon EC2 Service Limits
- What is Service Quotas?
- How to Request Quota Increase
- Service endpoints and quotas
- Guida per l'utente di Service Quotas
- Quota Monitor for AWS
- AWS Limiti di isolamento dei guasti di
- Availability with redundancy
- AWS for Data
- Cos'è l'integrazione continua?
- Cos'è la distribuzione continua?
- Partner APN: partner per la gestione della configurazione
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS
- Managing and monitoring API throttling in your workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower
- Actions, resources, and condition keys for Service Quotas

Video correlati:

- AWS Live re:Inforce 2019 - Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas

- AWS IAM Quotas Demo
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small

Strumenti correlati:

- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- Marketplace AWS

REL01-BP04 Monitoraggio e gestione delle quote

Valuta il tuo utilizzo potenziale e aumenta le quote in modo opportuno per una crescita pianificata dell'utilizzo.

Risultato desiderato: implementazione di sistemi attivi e automatizzati per la gestione e il monitoraggio. Queste soluzioni operative indicano che le soglie di utilizzo delle quote stanno per essere raggiunte. Il problema può essere risolto in modo proattivo tramite modifiche alle quote richieste.

Anti-pattern comuni:

- Mancata configurazione del monitoraggio per verificare le soglie delle quote di servizio.
- Mancata configurazione del monitoraggio dei limiti fissi, anche se i valori non possono essere modificati.
- Valutazione errata della quantità di tempo necessaria per richiedere e ottenere la modifica di una quota flessibile, supponendo che sia immediata o rapida.

- Configurazione di allarmi per l'avvicinamento alle quote di servizio, ma senza alcun processo di risposta a un avviso.
- Configurazione degli allarmi solo per i servizi supportati da Service AWS Quotas e non monitoraggio di altri servizi. AWS
- Valutazione errata della gestione delle quote per progettazioni di resilienza in più regioni, come gli approcci attivo/attivo, attivo/passivo con standby a caldo, attivo/passivo con standby a freddo e attivo/passivo con Pilot Light.
- Mancata valutazione delle differenze di quota tra regioni.
- Mancata valutazione delle esigenze in ogni regione per una richiesta di aumento di quota specifica.
- Mancato utilizzo di modelli per la gestione delle quote multiregioni.

Vantaggi derivanti dall'adozione di questa best practice: il monitoraggio automatico delle AWS Service Quotas e il monitoraggio dell'utilizzo rispetto a tali quote ti consentiranno di vedere quando ti stai avvicinando a un limite di quota. Puoi usare questi dati di monitoraggio per limitare eventuali errori dovuti all'esaurimento della quota.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Per i servizi supportati, puoi monitorare le quote configurando servizi diversi in grado di eseguire una valutazione e quindi inviare avvisi o allarmi. In questo modo, il monitoraggio dell'utilizzo è più semplice e puoi ricevere avvisi all'avvicinamento delle quote. Questi allarmi possono essere richiamati da, funzioni AWS Config Lambda, Amazon CloudWatch o da. AWS Trusted Advisor Puoi anche utilizzare i filtri metrici sui CloudWatch registri per cercare ed estrarre modelli nei log per determinare se l'utilizzo si avvicina alle soglie di quota.

Passaggi dell'implementazione

Per il monitoraggio:

- Acquisisci informazioni sull'attuale consumo di risorse, ad esempio bucket o istanze. Utilizza API le operazioni di servizio, come Amazon EC2 `DescribeInstances`API, per raccogliere l'attuale consumo di risorse.
- Acquisisci le attuali quote essenziali e valide per i servizi usando:
  - AWS Service Quotas

  - AWS Trusted Advisor
  - AWS documentazione
  - AWS pagine specifiche del servizio
  - AWS Command Line Interface (AWS CLI)
  - AWS Cloud Development Kit (AWS CDK)
- Utilizza AWS Service Quotas, un AWS servizio che ti aiuta a gestire le quote per oltre 250 AWS servizi da un'unica posizione.
- Utilizza i limiti Trusted Advisor di servizio per monitorare i tuoi attuali limiti di servizio a varie soglie.
- Utilizza la cronologia delle quote di servizio (console o AWS CLI) per verificare gli aumenti regionali.
- Confronta la modifica delle quote di servizio in ogni regione e ogni account per creare equivalenze, se necessario.

Per la gestione:

- Automatizzato: imposta una regola AWS Config personalizzata per scansionare le quote di servizio tra le regioni e confrontarle per individuare le differenze.
- Automatica: configura una funzione Lambda personalizzata per analizzare le quote di servizio tra regioni e confrontarle per individuare le differenze.
- Manuale: scansiona la quota dei servizi tramite AWS CLI o AWS Console per scansionare le quote di servizio tra le regioni e confrontarle per individuare eventuali differenze. API Segnala le differenze.
- In caso di individuazione di differenze nelle quote tra regioni, richiedi una modifica della quota, se necessario.
- Esamina il risultato di tutte le richieste.

Risorse

Best practice correlate:

- REL01-BP01 Consapevolezza su quote e vincoli di servizio
- REL01-BP02 Gestione delle quote di servizio in più account e regioni
- REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura

- REL01-BP05 Automazione della gestione delle quote
- REL01-BP06 Creazione di un divario sufficiente tra le quote attuali e l'utilizzo massimo per consentire eventuali failover
- REL03-BP01 Scegli come segmentare il tuo carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL12-BP04 Test della resilienza tramite l'utilizzo dell'ingegneria del caos

Documenti correlati:

- AWS Il pilastro dell'affidabilità di Well-Architected Framework: disponibilità
- AWS Service Quotas (precedentemente denominate limiti di servizio)
- AWS Trusted Advisor Controlli relativi alle migliori pratiche (vedere la sezione Limiti del servizio)
- AWS limita il monitoraggio delle AWS risposte
- Limiti EC2 del servizio Amazon
- What is Service Quotas?
- How to Request Quota Increase
- Service endpoints and quotas
- Guida per l'utente di Service Quotas
- Quota Monitor per AWS
- AWS Limiti di isolamento dei guasti
- Availability with redundancy
- AWS per i dati
- Cos'è l'integrazione continua?
- Cos'è la distribuzione continua?
- APNPartner: partner che possono aiutare nella gestione della configurazione
- Gestione del ciclo di vita dell'account in ambienti SaaS account-per-tenant su AWS
- Gestione e monitoraggio della limitazione dei carichi di API lavoro
- Visualizza i AWS Trusted Advisor consigli su larga scala con AWS Organizations

- Automatizzazione degli aumenti dei limiti di servizio e del supporto aziendale con AWS Control Tower
- Actions, resources, and condition keys for Service Quotas

Video correlati:

- AWS Live re:InForce 2019 - Service Quotas
- Visualizzazione e gestione delle quote per AWS i servizi tramite Service Quotas
- AWS IAMDimostrazione di Quotas
- AWS re:Invent 2018: Chiudere i circuiti e aprire le menti: come assumere il controllo di sistemi, grandi e piccoli

Strumenti correlati:

- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- Marketplace AWS

REL01-BP05 Automazione della gestione delle quote

Service Quotas, chiamate anche limiti nei servizi AWS, sono i valori massimi per le risorse dell'Account AWS. Ogni servizio AWS definisce un set di quote e i relativi valori predefiniti. Per fornire al carico di lavoro l'accesso a tutte le risorse necessarie, potrebbe essere necessario aumentare i valori di Service Quotas.

L'aumento del consumo delle risorse AWS da parte del carico di lavoro può minacciare la stabilità del carico di lavoro e avere un impatto sull'esperienza dell'utente in caso di superamento delle

quote. Implementa strumenti che segnalano quando il carico di lavoro si avvicina ai limiti e valuta la possibilità di creare automaticamente richieste di aumento delle quote.

Risultato desiderato: le quote sono configurate in modo appropriato per i carichi di lavoro in esecuzione in ciascun Account AWS e Regione.

Anti-pattern comuni:

- Non riesci a considerare e regolare le quote in modo appropriato per soddisfare i requisiti del carico di lavoro.
- Tieni traccia delle quote e dell'utilizzo mediante metodi che possono diventare obsoleti, come ad esempio i fogli di calcolo.
- Aggiorni i limiti di servizio solo in base a pianificazioni periodiche.
- L'organizzazione non dispone di processi operativi per rivedere le quote esistenti e richiedere aumenti di Service Quotas quando necessario.

Vantaggi dell'adozione di questa best practice:

- Maggiore resilienza del carico di lavoro: eviti gli errori causati dal superamento delle quote di risorse AWS.
- Disaster recovery semplificato: puoi riutilizzare i meccanismi di gestione automatica delle quote creati nella Regione primaria durante la configurazione del disaster recovery in un'altra Regione AWS.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Visualizza le quote correnti e tieni traccia del consumo di quote in corso attraverso meccanismi quali la console AWS Service Quotas, AWS Command Line Interface (AWS CLI) e gli AWS SDK. Inoltre, puoi integrare i database di gestione della configurazione (CMDB) e i sistemi di gestione dei servizi IT (ITSM) con le API di AWS Service Quotas.

Genera avvisi automatici se l'utilizzo delle quote raggiunge le soglie definite e definisci un processo per presentare richieste di aumento della quota quando ricevi avvisi. Se il carico di lavoro sottostante è critico per l'azienda, puoi automatizzare le richieste di aumento della quota, ma esegui il test dell'automazione con cautela per evitare il rischio di un'azione incontrollata, come un ciclo di feedback della crescita.

Gli aumenti di quota più piccoli vengono spesso approvati automaticamente. È possibile che per le richieste di quote più grandi sia richiesta l'elaborazione manuale a livello di Supporto AWS e che il tempo richiesto per la revisione e l'elaborazione sia maggiore. Prevedi un tempo aggiuntivo per l'elaborazione di più richieste o richieste di grandi incrementi.

Passaggi dell'implementazione

- Implementa il monitoraggio automatico di Service Quotas e invia avvisi se la crescita dell'utilizzo delle risorse del carico di lavoro si avvicina ai limiti delle quote. Ad esempio, Monitoraggio delle quote per AWS può fornire il monitoraggio automatico di Service Quotas. Questo strumento si integra con AWS Organizations e si distribuisce utilizzando Cloudformation StackSets in modo che i nuovi account vengano monitorati automaticamente al momento della creazione.
- Utilizza funzionalità come i modelli di richiesta Service Quotas o AWS Control Tower per semplificare la configurazione di Service Quotas per nuovi account.
- Crea dashboard dell'utilizzo attuale di Service Quotas in tutti gli Account AWS e le Regioni e fai riferimenti ad esse, se necessario, per evitare di superare le quote. La Dashboard Trusted Advisor Organizational (TAO), che fa parte delle Dashboard di cloud intelligence, permette di iniziare rapidamente a usare una dashboard di questo tipo.
- Tieni traccia delle richieste di aumento dei limiti di servizio. Consolidated Insights from Multiple Accounts (CIMA) può fornire una visione a livello di organizzazione di tutte le richieste.
- Verifica la generazione di avvisi e l'automazione delle richieste di aumento della quota impostando soglie di quota più basse negli account non di produzione. Non eseguire questi test in un account di produzione.

Risorse

Best practice correlate:

- OPS10-BP07 Automazione delle risposte agli eventi

Documenti correlati:

- Partner APN: partner per la gestione della configurazione
- Marketplace AWS: prodotti CMDB per il monitoraggio delle restrizioni
- AWS Service Quotas (precedentemente definite restrizioni dei servizi)
- AWS Trusted Advisor Best Practice Checks (consulta la sezione Service Limits)

- Quota Monitor Solution on AWS - AWS Solution
- What is Service Quotas?
- What is Service Quotas request templates?

Video correlati:

- AWS Live re:Inforce 2019 - Service Quotas
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower

Strumenti correlati:

- Quota Monitor for AWS

REL01-BP06 Creazione di un divario sufficiente tra le quote attuali e l'utilizzo massimo per consentire eventuali failover

Il presente articolo illustra come mantenere lo spazio tra la quota di risorse e l'utilizzo e i relativi vantaggi per la tua organizzazione. Una volta terminato l'utilizzo di una risorsa, la quota di utilizzo può continuare a essere conteggiata per tale risorsa, con possibile conseguenza di una risorsa in errore o inaccessibile. Evita tale errore nelle risorse verificando che le quote tengano conto della sovrapposizione di risorse in errore o inaccessibili e della rispettiva sostituzione. Prendi in considerazioni casi come errori della rete, errori della zona di disponibilità o errori della regione durante il calcolo di questo divario.

Risultato desiderato: è possibile coprire piccoli o grandi errori nelle risorse o nell'accessibilità delle risorse entro le attuali soglie di servizio, tenendo conto degli errori delle zone, di rete o addirittura regionali nella pianificazione delle risorse.

Anti-pattern comuni:

- Impostazione delle quote di servizio in base alle esigenze attuali senza tenere conto degli scenari di failover.
- Calcolo della quota massima per un servizio senza tenere conto dei principali della stabilità statica.
- Calcolo della quota totale necessaria per ogni regione senza tenere conto delle potenziali risorse inaccessibili.
- Valutazione errata dei limiti di isolamento degli errori per alcuni servizi AWS e dei possibili modelli di utilizzo anomalo.

Vantaggi dell'adozione di questa best practice: in caso di eventi di interruzione del servizio che influiscono sulla disponibilità dell'applicazione, utilizza il cloud per implementare strategie di ripristino da tali eventi. Un esempio di strategia consiste nella creazione di risorse aggiuntive per sostituire quelle inaccessibili e soddisfare le condizioni di failover senza esaurire il limite del servizio.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Nel valutare un limite di quota, tieni conto dei casi di failover che possono verificarsi a causa di un peggioramento della situazione. Considera i casi di failover seguenti:

- VPC interrotto o inaccessibile.
- Sottorete inaccessibile.
- Zona di disponibilità degradata che influisce sull'accessibilità delle risorse.
- Diversi instradamenti di rete o punti di ingresso e uscita bloccati o modificati.
- Impatto di una regione degradata sull'accessibilità delle risorse.
- Errore in un sottoinsieme di risorse in una regione o in una zona di disponibilità.

La decisione relativa all'avvio del failover è unica per ogni situazione, in quanto l'impatto aziendale può variare. Gestisci la pianificazione della capacità delle risorse nella posizione di failover e le quote delle risorse prima di decidere di effettuare il failover di un'applicazione o di un servizio.

Prendi in considerazione i picchi di attività più elevati del normale nell'esame delle quote per ciascun servizio. Questi picchi potrebbero essere correlati a risorse ancora attive ma inaccessibili a causa di reti o autorizzazioni. Le risorse attive non terminate vengono conteggiate rispetto al limite di quota del servizio.

Passaggi dell'implementazione

- Mantieni uno spazio sufficiente tra la quota di servizio e l'utilizzo massimo in modo da gestire un failover o la perdita di accessibilità.
- Determina le quote di servizio. Tieni conto di modelli di implementazione tipici, requisiti di disponibilità e crescita dei consumi.
- Richiedi aumenti delle quote, se necessario. Prevedi un tempo di attesa per la richiesta di aumento della quota.
- Determina i requisiti di affidabilità, noti anche come numero di 9.

- Analizza i potenziali scenari di errore, come la perdita di un componente, di una zona di disponibilità o di una regione.
- Stabilisci la metodologia di implementazione (ad esempio, canary, blu/verde, rosso/nero e rolling).
- Includi un buffer appropriato rispetto al limite della quota attuale. Un esempio di buffer potrebbe essere del 15%.
- Includi calcoli per la stabilità statica (zonale e regionale) laddove appropriato.
- Pianifica la crescita dei consumi e monitora i trend di consumo.
- Tieni conto dell'impatto della stabilità statica per i carichi di lavoro più critici. Valuta la conformità delle risorse a un sistema statisticamente stabile in tutte le regioni e le zone di disponibilità.
- Valuta l'utilizzo di prenotazioni della capacità on demand per pianificare la capacità in anticipo rispetto a qualsiasi failover. Si tratta di una strategia utile da implementare per le pianificazioni aziendali critiche per ridurre i possibili rischi legati all'ottenimento della quantità e del tipo di risorse corretti durante il failover.

Risorse

Best practice correlate:

- REL01-BP01 Consapevolezza su quote e vincoli di servizio
- REL01-BP02 Gestione delle quote di servizio in più account e regioni
- REL01-BP03 Adattamento di quote e vincoli di servizio fissi mediante l'architettura
- REL01-BP04 Monitoraggio e gestione delle quote
- REL01-BP05 Automazione della gestione delle quote
- REL03-BP01 Scegli come segmentare il tuo carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL12-BP04 Test della resilienza tramite l’utilizzo dell’ingegneria del caos

Documenti correlati:

- AWS Pilastro dell’affidabilità di Well-Architected Framework: disponibilità
- AWS Service Quotas (precedentemente definite restrizioni dei servizi)

- AWS Trusted Advisor Best Practice Checks (consulta la sezione Service Limits)
- AWS limit monitor on AWS answers
- Amazon EC2 Service Limits
- What is Service Quotas?
- How to Request Quota Increase
- Service endpoints and quotas
- Guida per l'utente di Service Quotas
- Quota Monitor for AWS
- Limiti di isolamento dei guasti di AWS
- Availability with redundancy
- AWS for Data
- Cos'è l'integrazione continua?
- Cos'è la distribuzione continua?
- Partner APN: partner per la gestione della configurazione
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS
- Managing and monitoring API throttling in your workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower
- Actions, resources, and condition keys for Service Quotas

Video correlati:

- AWS Live re:Inforce 2019 - Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas
- AWS IAM Quotas Demo
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small

Strumenti correlati:

- AWS CodeDeploy
- AWS CloudTrail

- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- Marketplace AWS

REL 2. Come si pianifica la topologia della rete?

I carichi di lavoro sono spesso presenti in più ambienti. Questi includono più ambienti cloud (sia accessibili pubblicamente sia privati) e, possibilmente, l'infrastruttura del data center esistente. I piani devono includere considerazioni di rete, ad esempio connettività intrasistema e intersistema, gestione di indirizzi IP pubblici, gestione di indirizzi IP privati e risoluzione dei nomi di dominio.

Best practice

- REL02-BP01 Utilizzo di una connettività di rete a disponibilità elevata per gli endpoint pubblici del carico di lavoro
- REL02-BP02 Esecuzione del provisioning di connettività ridondante tra reti private nel cloud e negli ambienti on-premises
- REL02-BP03 Verifica che l'allocazione delle sottoreti IP consenta l'espansione e la disponibilità
- REL02-BP04 Preferire topologie hub-and-spoke rispetto a mesh da-molti-a-molti
- REL02-BP05 Applicazione di intervalli di indirizzi IP privati non sovrapposti in tutti gli spazi con indirizzi privati a cui sono connessi

REL02-BP01 Utilizzo di una connettività di rete a disponibilità elevata per gli endpoint pubblici del carico di lavoro

La creazione di connettività di rete a disponibilità elevata agli endpoint pubblici dei carichi di lavoro può ridurre i tempi di inattività dovuti a perdita di connettività e migliorare la disponibilità e il contratto sul livello di servizio del tuo carico di lavoro. Per ottenere questo risultato, usa un servizio DNS a disponibilità elevata, reti di distribuzione di contenuti (CDN), API Gateway, bilanciamento del carico o proxy inversi.

Risultato desiderato: la pianificazione, la realizzazione e la messa in funzione di una connettività di rete altamente disponibile per i tuoi endpoint pubblici è fondamentale. Se il carico di lavoro diventa irraggiungibile a causa della perdita di connettività, il sistema apparirà ai clienti come non funzionante, anche se il carico di lavoro è in esecuzione e disponibile. Combinando connettività di rete a disponibilità elevata e resiliente per gli endpoint pubblici del carico di lavoro, a un'architettura resiliente per il carico di lavoro stesso, puoi offrire ai clienti la disponibilità e il livello di servizio migliori possibili.

AWS Global Accelerator, Amazon CloudFront, Gateway Amazon API, funzione URL AWS Lambda, API AWS AppSync ed Elastic Load Balancing (ELB) forniscono tutti endpoint pubblici a elevata disponibilità. Amazon Route 53 fornisce un servizio DNS ad alta disponibilità per la risoluzione dei nomi di dominio, così da verificare la possibilità di risolvere gli indirizzi degli endpoint pubblici.

Puoi anche valutare applicazioni software Marketplace AWS per il bilanciamento del carico e l'esecuzione di proxy.

Anti-pattern comuni:

- Progettazione di un carico di lavoro a disponibilità elevata senza pianificare connettività DNS e di rete per la disponibilità elevata.
- Uso di indirizzi Internet pubblici su singoli container o istanze e gestione della connettività tramite DNS.
- Uso di indirizzi IP anziché nomi di dominio per l'individuazione dei servizi.
- Mancata esecuzione di test su scenari con perdita di connettività agli endpoint pubblici.
- Mancata analisi delle esigenze di throughput della rete e dei modelli di distribuzione.
- Nessuna attività di test e pianificazione per scenari di possibile interruzione della connettività di rete Internet agli endpoint pubblici del carico di lavoro.
- Distribuzione di contenuti (pagine Web, asset statici o file multimediali) in un'area geografica di grandi dimensioni senza l'uso di una rete di distribuzione di contenuti.
- Nessuna pianificazione per la prevenzione di attacchi DDoS (Distributed Denial of Service). Gli attacchi DDoS rischiano di arrestare il traffico legittimo e ridurre la disponibilità per gli utenti.

Vantaggi dell'adozione di questa best practice: la progettazione pensata per una connettività di rete a elevata disponibilità e resilienza garantisce l'accessibilità e la disponibilità del carico di lavoro agli utenti.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Alla base della creazione di connettività di rete a disponibilità elevata agli endpoint pubblici vi è l'instradamento del traffico. Per verificare che il traffico possa raggiungere gli endpoint, il servizio DNS deve essere in grado di risolvere i nomi di dominio negli indirizzi IP corrispondenti. Utilizza un sistema dei nomi di dominio (DNS) altamente scalabile e disponibile, come Amazon Route 53, per gestire i record DNS del dominio. Puoi usare anche i controlli dell'integrità forniti da Amazon Route 53. I controlli dell'integrità verificano che l'applicazione sia raggiungibile, disponibile e funzionale e possono essere configurati in modo da simulare il comportamento degli utenti, come la richiesta di una pagina Web o un URL specifico. In caso di errore, Amazon Route 53 risponde alle richieste di risoluzione DNS e indirizza il traffico solo agli endpoint integri. Puoi anche valutare se usare le funzionalità di instradamento basato sulla latenza e GeoDNS offerte da Amazon Route 53.

Per verificare l'elevata disponibilità effettiva del carico di lavoro, utilizza Elastic Load Balancing (ELB). Amazon Route 53 consente di indirizzare il traffico verso ELB, che lo distribuisce alle istanze di calcolo di destinazione. Puoi anche usare Gateway Amazon API insieme a AWS Lambda per una soluzione serverless. I clienti possono anche eseguire carichi di lavoro in più Regioni AWS. Grazie a un pattern attivo/attivo multisito, il carico di lavoro può servire il traffico proveniente da più regioni. Con un pattern attivo/passivo multisito, il carico di lavoro serve il traffico proveniente dalla regione attiva, mentre nella regione secondaria avviene la replica dei dati, che diventano attivi in caso di guasto nella regione primaria. I controlli dell'integrità di Route 53 consentono dunque di controllare il failover DNS da qualsiasi endpoint in una regione primaria a un endpoint in una regione secondaria, verificando la raggiungibilità e la disponibilità del carico di lavoro per gli utenti.

Amazon CloudFront offre una semplice API per la distribuzione di contenuti con bassa latenza e velocità di trasferimento dati elevate gestendo le richieste tramite una rete di posizioni edge in tutto il mondo. Le reti di distribuzione di contenuti (CDN) operano per i clienti, distribuendo i contenuti situati o memorizzati nella cache in una posizione vicina all'utente. In questo modo si migliora anche la disponibilità dell'applicazione poiché il carico dei contenuti viene spostato dai server alle posizioni edge di CloudFront. Le posizioni edge e le cache edge regionali includono copie memorizzate nella cache del contenuto vicino agli utenti, per il recupero rapido e una raggiungibilità e una disponibilità maggiori del carico di lavoro.

Per i carichi di lavoro con utenti distribuiti in più aree geografiche, AWS Global Accelerator contribuisce a migliorare la disponibilità e le prestazioni delle applicazioni. AWS Global Accelerator fornisce indirizzi IP statici anycast che operano come punto di ingresso statico alle applicazioni ospitate in una o più Regioni AWS. In questo modo, il traffico può entrare nella rete globale AWS il più vicino possibile agli utenti, migliorando così la raggiungibilità e la disponibilità del carico di

lavoro. AWS Global Accelerator monitora anche l'integrità degli endpoint dell'applicazione usando controlli dell'integrità TCP, HTTP e HTTPS. Eventuali variazioni dell'integrità o della configurazione degli endpoint permettono il reindirizzamento del traffico degli utenti a endpoint integri che offrono le prestazioni e la disponibilità migliori agli utenti. Inoltre, AWS Global Accelerator presenta una progettazione di isolamento degli errori che usa due indirizzi IPv4 statici gestiti da zone di rete indipendenti, migliorando la disponibilità delle applicazioni.

Per proteggere i clienti dagli attacchi DDoS, AWS offre AWS Shield Standard. Shield Standard si attiva in automatico e protegge dagli attacchi comuni all'infrastruttura (livello 3 e 4) come i flood SYN/UDP e gli attacchi di riflessione in modo da supportare l'elevata disponibilità delle applicazioni su AWS. Per altre soluzioni di protezione da attacchi più sofisticati e di maggiore entità (come i flood UDP) e di tipo state-exhaustion (come i flood TCP SYN) e per proteggere le applicazioni in esecuzione su Amazon Elastic Compute Cloud (Amazon EC2), Elastic Load Balancing (ELB), Amazon CloudFront, AWS Global Accelerator e Route 53, puoi prendere in considerazione l'uso di AWS Shield Advanced. Per la protezione da attacchi a livello di applicazione come i flood HTTP POST o GET, usa AWS WAF. AWS WAF può usare indirizzi IP, intestazioni HTTP, corpo HTTP, stringhe URI, iniezione SQL e condizioni di scripting cross-site per determinare se una richiesta debba essere bloccata o consentita.

Passaggi dell'implementazione

1. Configura DNS a elevata disponibilità: Amazon Route 53 è un servizio Web di sistema dei nomi di dominio (DNS) altamente scalabile e disponibile. Route 53 collega le richieste degli utenti alle applicazioni Internet eseguite su AWS oppure on-premises. Per ulteriori informazioni, consulta configuring Amazon Route 53 as your DNS service.
2. Configura controlli dell'integrità: quando usi Route 53, verifica che solo le destinazioni integre siano risolvibili. Inizia con la creazione dei controlli dell'integrità di Route 53 e la configurazione del failover DNS. Nel configurare controlli dell'integrità, è importante tenere conto degli aspetti seguenti:
    a. Modo in cui Amazon Route 53 determina se un controllo dell'integrità ha esito positivo
    b. Creazione, aggiornamento ed eliminazione di controlli dell'integrità
    c. Monitoraggio dello stato dei controlli dell'integrità e ricezione di notifiche
    d. Best practice per Amazon Route 53 DNS
3. Connessione del servizio DNS agli endpoint.
    a. In caso di utilizzo di Elastic Load Balancing come target per il tuo traffico, crea un record di alias mediante Amazon Route 53 che punti all'endpoint regionale del tuo sistema bilanciatore del

carico. Durante la creazione del record di alias, imposta l'opzione Valutazione dello stato target su Sì.

b. In caso di utilizzo di API Gateway, per i carichi di lavoro serverless o le API private, usa Route 53 per indirizzare il traffico verso l'API Gateway.

4. Opta per una rete di distribuzione di contenuti (CDN).

   a. Per la distribuzione di contenuti mediante posizioni edge più vicine all'utente, esamina il modo in cui CloudFront distribuisce i contenuti.

   b. Inizia partendo con una distribuzione CloudFront semplice. CloudFront sa quindi determinare dove vuoi distribuire i contenuti e come monitorare e gestire la distribuzione di contenuti. Nel configurare la distribuzione di CloudFront, è importante tenere conto degli aspetti seguenti:

      i. Come funziona la memorizzazione nella cache con le posizioni edge di CloudFront

      ii. Aumento della percentuale di richieste eseguite direttamente dalle cache CloudFront (percentuale di riscontri nella cache)

      iii. Utilizzo dello scudo di origine Amazon CloudFront

      iv. Ottimizzazione dell'elevata disponibilità con il failover di origine CloudFront

5. Configura la protezione a livello di applicazione: AWS WAF semplifica la protezione da exploit Web e bot comuni che possono compromettere la disponibilità e la sicurezza o consumare risorse eccessive. Per una conoscenza più approfondita, scopri come funziona AWS WAF e quando sarà tutto pronto per implementare le protezioni dai flood HTTP POST e GET a livello dell'applicazione, consulta Getting started with AWS WAF. Puoi anche utilizzare AWS WAF con CloudFront. Consulta la documentazione su come funziona AWS WAF con le funzionalità di Amazon CloudFront.

6. Configura protezione aggiuntiva da attacchi DDoS: per impostazione predefinita, tutti i clienti AWS ricevono protezione gratuita dagli attacchi DDoS comuni e più frequenti a livello di rete e di trasporto che prendono di mira il sito Web o l'applicazione con AWS Shield Standard. Per una protezione aggiuntiva delle applicazioni con accesso a Internet in esecuzione su Amazon EC2, Elastic Load Balancing, Amazon CloudFront, AWS Global Accelerator e Amazon Route 53, puoi prendere in considerazione AWS Shield Advanced ed esaminare gli esempi di architetture resilienti agli attacchi DDoS. Per proteggere carico di lavoro ed endpoint pubblici dagli attacchi DDoS, consulta Getting started with AWS Shield Advanced.

Risorse

Best practice correlate:

- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP04 Fare affidamento al piano dati invece che al piano di controllo (control-plane) durante il ripristino
- REL11-BP06 Invio di notifiche quando gli eventi influiscono sulla disponibilità

Documenti correlati:

- Partner APN: partner per la pianificazione della rete
- Marketplace AWS per l'infrastruttura di rete
- Cosa è AWS Global Accelerator?
- What is Amazon CloudFront?
- What is Amazon Route 53?
- Cos'è l'Elastic Load Balancing?
- Network Connectivity capability - Establishing Your Cloud Foundations
- What is Amazon API Gateway?
- What are AWS WAF, AWS Shield, and AWS Firewall Manager?
- What is Amazon Application Recovery Controller?
- Configure custom health checks for DNS failover

Video correlati:

- AWS re:Invent 2022 - Improve performance and availability with AWS Global Accelerator
- AWS re:Invent 2020: Global traffic management with Amazon Route 53
- AWS re:Invent 2022 - Operating highly available Multi-AZ applications
- AWS re:Invent 2022 - Dive deep on AWS networking infrastructure
- AWS re:Invent 2022 - Dive deep on networking infrastructure

Esempi correlati:

- Disaster Recovery with Amazon Application Recovery Controller (ARC)
- AWS Global Accelerator Workshop

REL02-BP02 Esecuzione del provisioning di connettività ridondante tra reti private nel cloud e negli ambienti on-premises

Implementa la ridondanza delle connessioni tra reti private nel cloud e negli ambienti on-premises per ottenere la resilienza della connettività. A tal fine, puoi implementare due o più collegamenti e percorsi di traffico, preservando la connettività in caso di errori di rete.

Anti-pattern comuni:

- Dipendi da una sola connessione di rete, che crea un singolo punto di errore.
- Utilizzi un solo tunnel VPN o più tunnel che terminano nella stessa zona di disponibilità.
- Ti affidi a un solo ISP per la connettività VPN, suscettibile di guasto totale in caso di interruzione dell'ISP.
- Non implementi i protocolli di instradamento dinamico come BGP, fondamentali per reindirizzare il traffico durante le interruzioni di rete.
- Ignori i limiti di larghezza di banda dei tunnel VPN e sopravvaluti le capacità di backup.

Vantaggi dell'adozione di questa best practice: implementando una connettività ridondante tra il tuo ambiente cloud e l'ambiente aziendale oppure on-premises, puoi garantire l'affidabilità delle comunicazioni dei servizi dipendenti tra due ambienti.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Quando si utilizza AWS Direct Connect per connettere la rete on-premises ad AWS, è possibile ottenere la massima resilienza di rete (SLA del 99,99%) impiegando connessioni separate che terminano su dispositivi diversi in più di una posizione on-premises e in più di una posizione AWS Direct Connect. Questa topologia offre resilienza ai guasti dei dispositivi, ai problemi di connettività e alle interruzioni complete della posizione. In alternativa, puoi ottenere un'elevata resilienza (SLA del 99,9%) utilizzando due singole connessioni a più posizioni, con ciascuna posizione on-premises connessa a una singola posizione Direct Connect. Questo approccio offre protezione dalle interruzioni della connettività causate da interruzioni della fibra o guasti dei dispositivi e aiuta a mitigare le interruzioni complete della posizione. Il kit di strumenti di resilienza di Direct Connect può aiutarti a progettare la tua topologia AWS Direct Connect.

Puoi anche prendere in considerazione l'utilizzo di AWS Site-to-Site VPN che termina su AWS Transit Gateway come soluzione conveniente di backup sulla connessione primaria AWS Direct Connect.

Questa configurazione abilita l'instradamento equal-cost multi-path (ECMP) su più tunnel VPN, consentendo un throughput fino a 50 Gbps, anche se ogni tunnel VPN è limitato a 1,25 Gbps. È importante notare, tuttavia, che AWS Direct Connect è ancora la scelta più efficace per ridurre al minimo le interruzioni di rete e garantire una connettività stabile.

Quando utilizzi le VPN su Internet per connettere l'ambiente cloud al tuo data center on-premises, configura due tunnel VPN come parte di un'unica connessione VPN sito-sito. Ogni tunnel deve terminare in una zona di disponibilità diversa per garantire l'alta disponibilità e utilizzare hardware ridondante per prevenire gli errori dei dispositivi on-premises. Inoltre, prendi in considerazione l'uso di più connessioni Internet di vari provider di servizi Internet (ISP) per la tua posizione on-premises per evitare l'interruzione completa della connettività VPN dovuta al guasto di un singolo ISP. La scelta di ISP con instradamento e infrastrutture diversi, in particolare quelli con percorsi fisici separati verso gli endpoint AWS, offre un'elevata disponibilità della connettività.

Oltre alla ridondanza fisica con più connessioni AWS Direct Connect e più tunnel VPN, o una combinazione di entrambi, è fondamentale anche l'implementazione dell'instradamento dinamico del Border Gateway Protocol (BGP). Il BGP dinamico fornisce il reinstradamento automatico del traffico da un percorso all'altro in base alle condizioni della rete in tempo reale e alle policy configurate. Questo comportamento dinamico è particolarmente utile per mantenere la disponibilità della rete e la continuità del servizio in caso di errori di collegamento o rete. Seleziona rapidamente percorsi alternativi, migliorando la resilienza e l'affidabilità della rete.

Passaggi dell'implementazione

- Acquisisci la connettività ad alta disponibilità tra AWS e l'ambiente on-premises.
  - Utilizza più connessioni AWS Direct Connect o tunnel VPN tra reti private implementate separatamente.
  - Utilizza più posizioni Direct Connect per ottenere un'elevata disponibilità.
  - Se utilizzi più Regioni AWS, garantisci la ridondanza in almeno due di esse.
- Utilizza AWS Transit Gateway, quando possibile, per terminare la connessione VPN.
- Valuta le appliance di Marketplace AWS per terminare le VPN o estendere la tua SD-WAN su AWS. Se utilizzi appliance di Marketplace AWS, distribuisci le istanze ridondanti per la disponibilità elevata in diverse zone di disponibilità.
- Fornisci una connessione ridondante all'ambiente on-premises.
  - Per soddisfare le esigenze di disponibilità, possono essere necessarie connessioni ridondanti a più Regioni AWS.
  - Usa l'Direct Connect Resiliency Toolkit per iniziare.

Risorse

Documenti correlati:

- AWS Direct Connect Resiliency Recommendations
- Using Redundant Site-to-Site VPN Connections to Provide Failover
- Routing policies and BGP communities
- Active/Active and Active/Passive Configurations in AWS Direct Connect
- Partner APN: partner per la pianificazione della rete
- Marketplace AWS per l'infrastruttura di rete
- Amazon Virtual Private Cloud Connectivity Options Whitepaper
- Realizzazione di un'infrastruttura di reti multi-VPC sicura e scalabile
- Using redundant Site-to-Site VPN connections to provide failover
- Using the Direct Connect Resiliency Toolkit to get started
- VPC Endpoints and VPC Endpoint Services (AWS PrivateLink)
- What Is Amazon VPC?
- What is a transit gateway?
- Cos'è AWS Site-to-Site VPN?
- Working with Direct Connect gateways

Video correlati:

- AWS re:Invent 2018: Advanced VPC Design and New Capabilities for Amazon VPC
- AWS re:Invent 2019: AWS Transit Gateway reference architectures for many VPCs

REL02-BP03 Verifica che l'allocazione delle sottoreti IP consenta l'espansione e la disponibilità

Gli intervalli di indirizzi IP di Amazon VPC devono essere abbastanza grandi da soddisfare i requisiti del carico di lavoro, tra cui la fattorizzazione nella futura espansione e l'allocazione di indirizzi IP alle sottoreti nelle zone di disponibilità. Ciò comprende bilanciatori del carico, istanze EC2 e applicazioni basate su container.

Quando si pianifica la topologia di rete, il primo passo è definire lo spazio stesso degli indirizzi IP. Gli intervalli di indirizzi IP privati (secondo le linee guida RFC 1918) dovrebbero essere allocati per ogni VPC. Nell'ambito di questo processo, soddisfa i seguenti requisiti:

- Lascia spazio per indirizzi IP per più di un VPC per regione.
- All'interno di un VPC, lascia spazio per più sottoreti affinché coprano più zone di disponibilità.
- Prendi in considerazione di lasciare spazio per un blocco CIDR inutilizzato all'interno di un VPC per un'espansione futura.
- Assicurati che sia disponibile spazio per gli indirizzi IP, al fine di soddisfare le esigenze di qualsiasi parco istanze EC2 transitorio che puoi utilizzare, ad esempio parchi istanze spot per il machine learning, cluster Amazon EMR o cluster Amazon Redshift. Una considerazione analoga andrebbe fatta per i cluster Kubernetes, come Amazon Elastic Kubernetes Service (Amazon EKS), poiché per impostazione predefinita a ciascun pod Kubernetes viene assegnato un indirizzo instradabile dal blocco CIDR VPC.
- Tieni presente che i primi quattro indirizzi IP e l'ultimo indirizzo IP in ogni blocco CIDR della sottorete sono riservati e non disponibili per l'uso.
- Tieni presente che il blocco CIDR VPC iniziale allocato al VPC non può essere modificato o eliminato, ma puoi aggiungere ulteriori blocchi CIDR non sovrapposti al VPC. I CIDR IPv4 della sottorete non possono essere modificati, mentre ciò è possibile con i CIDR IPv6.
- Il blocco CIDR VPC più grande possibile è /16 e il più piccolo è /28.
- Prendi in considerazione altre reti connesse (VPC, on-premises o altri provider cloud) e assicurati che lo spazio degli indirizzi IP non si sovrapponga. Per ulteriori informazioni, consulta REL02-BP05 Applicazione di intervalli di indirizzi IP privati non sovrapposti in tutti gli spazi con indirizzi privati a cui sono connessi.

Risultato desiderato: una sottorete IP scalabile può aiutarti a far fronte alla crescita futura e a evitare inutili sprechi.

Anti-pattern comuni:

- Mancata presa in considerazione della crescita futura, con conseguenti blocchi CIDR troppo piccoli e che richiedono una riconfigurazione, il che comporta tempi di inattività.
- Stima erronea del numero di indirizzi IP utilizzabili da un bilanciatore del carico elastico.
- Distribuzione di numerosi bilanciatori del carico a traffico elevato nelle stesse sottoreti.
- Utilizzo di meccanismi di dimensionamento automatico senza monitorare il consumo di indirizzi IP.
- Definizione di intervalli CIDR eccessivamente ampi ben oltre le aspettative di crescita futura, il che può portare a difficoltà di peering con altre reti con intervalli di indirizzi sovrapposti.

Vantaggi dell'adozione di questa best practice: in questo modo puoi consentire la crescita dei carichi di lavoro e continuare a fornire disponibilità nell'aumentare verticalmente.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Pianifica la tua rete in base a crescita, compliance normativa e integrazione con altre reti. Senza una pianificazione adeguata, la crescita può essere sottovalutata, la compliance normativa può cambiare e l'implementazione di acquisizioni o di connessioni a reti private può rivelarsi difficile.

- Seleziona gli Account AWS e le regioni pertinenti in base ai tuoi requisiti di servizio, di latenza, normativi e di disaster recovery (DR).
- Identifica le esigenze delle implementazioni di VPC regionali.
- Identifica le dimensioni dei VPC.
  - Stabilisci se intendi implementare connettività multi-VPC.
    - What Is a Transit Gateway?
    - Connettività multi-VPC a singola regione
  - Stabilisci se hai bisogno della segregazione delle reti a causa di requisiti normativi.
  - Crea VPC con blocchi CIDR di dimensioni adeguate per soddisfare le tue esigenze attuali e future.
    - Se non hai definito proiezioni di crescita, potresti preferire blocchi CIDR più grandi per ridurre il potenziale di riconfigurazione futura
  - Prendi in considerazione l'utilizzo di un indirizzo IPv6 per le sottoreti nell'ambito di VPC dual-stack. Un indirizzo IPv6 è adatto per l'uso in sottoreti private contenenti parchi istanze o contenitori temporanei che altrimenti richiederebbero un numero elevato di indirizzi IPv4.

Risorse

Best practice Well-Architected correlate:

- REL02-BP05 Applicazione di intervalli di indirizzi IP privati non sovrapposti in tutti gli spazi con indirizzi privati a cui sono connessi

Documenti correlati:

- Partner APN: partner per la pianificazione della rete

- Marketplace AWS per l'infrastruttura di rete
- Amazon Virtual Private Cloud Connectivity Options Whitepaper
- Multiple data center HA network connectivity
- Connettività multi-VPC a singola regione
- What Is Amazon VPC?
- IPv6 su AWS
- IPv6 on reference architectures
- Amazon Elastic Kubernetes Service launches IPv6 support
- Consigli per il tuo VPC - Classic Load Balancer
- Sottoreti delle Zone di disponibilità - Application Load Balancer
- Zone di disponibilità - Network Load Balancer

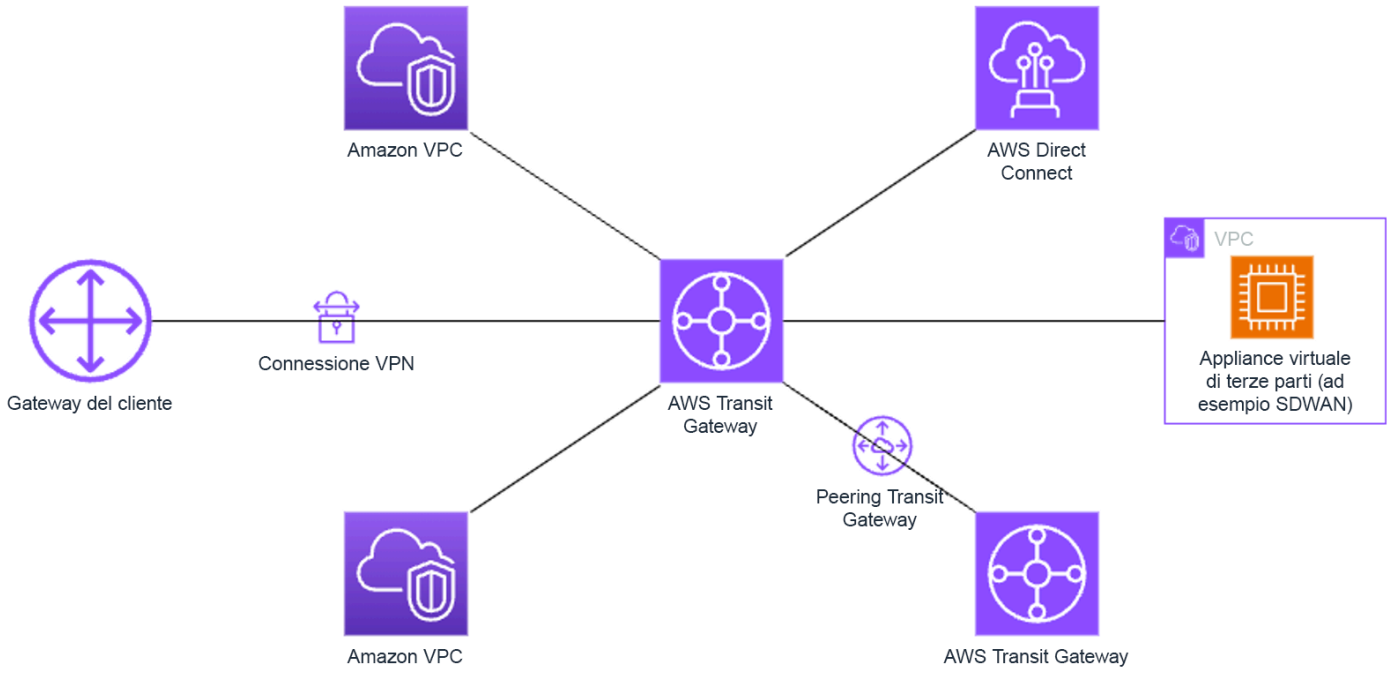
Video correlati:

- AWS re:Invent 2018: Advanced VPC Design and New Capabilities for Amazon VPC (NET303)
- AWS re:Invent 2019: AWS Transit Gateway reference architectures for many VPCs (NET406-R1)
- AWS re:Invent 2023: AWS Ready for what's next? Designing networks for growth and flexibility (NET310)

REL02-BP04 Preferire topologie hub-and-spoke rispetto a mesh da-molti-a-molti

Quando connetti più reti private, come cloud privati virtuali (VPC) e reti on-premises, è opportuno scegliere una topologia hub-and-spoke rispetto a una mesh. A differenza delle topologie mesh, in cui ogni rete si connette direttamente alle altre e aumenta la complessità e il sovraccarico di gestione, l'architettura hub-and-spoke centralizza le connessioni tramite un unico hub. Questa centralizzazione semplifica la struttura della rete e ne migliora il funzionamento, la scalabilità e il controllo.

AWS Transit Gateway è un servizio gestito, scalabile e a disponibilità elevata progettato per la creazione di reti hub-and-spoke su AWS. Funge da hub centrale della rete che fornisce la segmentazione, il routing centralizzato e la connessione semplificata agli ambienti cloud e on-premises. La figura seguente illustra come è possibile utilizzare AWS Transit Gateway per creare la topologia hub-and-spoke.



Risultato desiderato: hai connesso i cloud privati virtuali (VPC) e le reti on-premises tramite un hub centrale. Configuri le connessioni in peering tramite l'hub, che funge da router cloud a scalabilità elevata. L'instradamento è semplificato perché non è necessario lavorare con complesse relazioni di peering. Il traffico tra le reti è crittografato ed è possibile isolare le reti.

Anti-pattern comuni:

- Crei regole di peering di rete complesse.
- Fornisci instradamenti tra reti che non devono comunicare tra loro (ad esempio, carichi di lavoro separati che non hanno interdipendenze).
- La governance dell'istanza dell'hub è inefficace.

Vantaggi dell'adozione di questa best practice: con l'aumento del numero di reti connesse, la gestione e l'espansione della connettività mesh diventa sempre più impegnativa. Un'architettura mesh introduce ulteriori sfide, come componenti aggiuntivi dell'infrastruttura, requisiti di configurazione e considerazioni sull'implementazione. La mesh introduce inoltre costi operativi aggiuntivi per gestire e monitorare i componenti del piano dati e del piano di controllo (control-plane). Devi pensare a come fornire disponibilità elevata dell'architettura mesh, monitorare l'integrità e le prestazioni della mesh e gestire gli aggiornamenti dei componenti della mesh.

Un modello hub-and-spoke, invece, stabilisce un instradamento centralizzato del traffico su più reti. Consente un approccio più semplice alla gestione e al monitoraggio dei componenti del piano dati e del piano di controllo (control-plane).

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Crea un account dei Servizi di rete se non esiste. Posiziona l'hub nell'account dei Servizi di rete dell'organizzazione. Questo approccio consente ai tecnici della rete di gestire centralmente l'hub.

L'hub del modello hub-and-spoke funge da router virtuale per il traffico che scorre tra i cloud privati virtuali (VPC) e le reti on-premises. Questo approccio riduce la complessità della rete e facilita la risoluzione dei problemi di rete.

Considera la progettazione della rete, inclusi VPC, AWS Direct Connect e connessioni VPN da sito a sito che desideri interconnettere.

Considera l'utilizzo di una sottorete separata per ciascun collegamento VPC del gateway di transito. Per ogni sottorete, utilizza un piccolo CIDR (ad esempio /28), in modo da avere più spazio di indirizzi per le risorse di elaborazione. Inoltre, crea una lista di controllo degli accessi alla rete e associala a tutte le sottoreti collegate all'hub. Mantieni aperta la lista di controllo degli accessi di rete in entrata e in uscita.

Progetta e implementa le tabelle di routing in modo che gli instradamenti siano forniti solo tra le reti che devono comunicare. Ometti gli instradamenti tra reti che non devono comunicare tra loro (ad esempio, tra carichi di lavoro separati che non hanno interdipendenze).

Passaggi dell'implementazione

1. Pianifica la rete. Determina le reti che desideri connettere e verifica che non condividano intervalli CIDR sovrapposti.
2. Crea un AWS Transit Gateway e collega i VPC.
3. Se necessario, crea connessioni VPN o gateway Direct Connect e associali a Transit Gateway.
4. Definisci come viene instradato il traffico tra i VPC connessi e altre connessioni tramite la configurazione delle tabelle di routing di Transit Gateway.
5. Usa Amazon CloudWatch per monitorare e modificare come necessario le configurazioni per l'ottimizzazione delle prestazioni e dei costi.

Risorse

Best practice correlate:

- REL02-BP03 Verifica che l'allocazione delle sottoreti IP consenta l'espansione e la disponibilità
- REL02-BP05 Applicazione di intervalli di indirizzi IP privati non sovrapposti in tutti gli spazi con indirizzi privati a cui sono connessi

Documenti correlati:

- What Is a Transit Gateway?
- Transit gateway design best practices
- Realizzazione di un'infrastruttura di reti multi-VPC sicura e scalabile
- Building a global network using AWS Transit Gateway Inter-Region peering
- Opzioni di connettività di Amazon Virtual Private Cloud
- Partner APN: partner per la pianificazione della rete
- Marketplace AWS per l'infrastruttura di rete

Video correlati:

- AWS re:Invent 2023 - AWS networking foundations
- AWS re:Invent 2023 - Advanced VPC designs and new capabilities

Workshop correlati:

- Workshop su AWS Transit Gateway

REL02-BP05 Applicazione di intervalli di indirizzi IP privati non sovrapposti in tutti gli spazi con indirizzi privati a cui sono connessi

Gli intervalli di indirizzi IP di ogni VPC non devono sovrapporsi quando sono collegati in peering o connessi tramite Transit Gateway o VPN. Evita i conflitti di indirizzi IP tra VPC e ambienti on-premises o altri provider di servizi cloud utilizzati. Bisogna inoltre disporre di una soluzione per allocare gli intervalli di indirizzi IP privati quando necessario. Un sistema di gestione indirizzi IP (IPAM) può aiutarti ad automatizzare l'allocazione.

Risultato desiderato:

- Nessun conflitto di intervalli di indirizzi IP tra VPC, ambienti on-premises o altri provider di servizi cloud.
- La corretta gestione degli indirizzi IP consente di scalare più facilmente l'infrastruttura di rete per supportare la crescita e i cambiamenti dei requisiti di rete.

Anti-pattern comuni:

- Utilizzo nel VPC dello stesso intervallo di indirizzi IP usato on-premises, nella rete aziendale o in altro provider di servizi cloud.
- Mancato monitoraggio degli intervalli IP dei VPC utilizzati per distribuire i carichi di lavoro.
- Ricorso a processi manuali di gestione degli indirizzi IP, come i fogli di calcolo.
- Utilizzo di blocchi CIDR sovradimensionati o sottodimensionati, con conseguente spreco di indirizzi IP o spazio di indirizzi insufficiente per il carico di lavoro.

Vantaggi dell'adozione di questa best practice: la pianificazione attiva della rete garantisce di non avere più occorrenze dello stesso indirizzo IP nelle reti interconnesse. In questo modo si evitano problemi di instradamento in parti del carico di lavoro che utilizzano le diverse applicazioni.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Utilizza un sistema IPAM, come Amazon VPC IP Address Manager, per il monitoraggio e la gestione dell'utilizzo di CIDR. Su Marketplace AWS sono disponibili anche diversi IPAM. Valuta il tuo utilizzo potenziale su AWS, aggiungi intervalli CIDR ai VPC esistenti e crea i VPC per consentire la crescita pianificata dell'utilizzo.

Passaggi dell'implementazione

- Misura il consumo attuale del CIDR, ad esempio VPC e sottoreti.
  - Utilizza le operazioni delle API di servizi per raccogliere il consumo attuale di CIDR.
  - Sfrutta Amazon VPC IP Address Manager per individuare le risorse.
- Misura l'utilizzo attuale delle sottoreti.
  - Utilizza le operazioni delle API di servizio per raccogliere le sottoreti per VPC in ogni regione.
  - Sfrutta Amazon VPC IP Address Manager per individuare le risorse.

- Registra l'uso attuale.
- Verifica se hai creato intervalli di indirizzi IP sovrapposti.
- Calcola la capacità inutilizzata.
- Individua gli intervalli di indirizzi IP sovrapposti. Puoi effettuare la migrazione verso una nuova gamma di indirizzi o prendere in considerazione l'utilizzo di tecniche come gateway NAT privato o AWS PrivateLink se devi connettere gli intervalli sovrapposti.

Risorse

Best practice correlate:

- Protezione delle reti

Documenti correlati:

- Partner APN: partner per la pianificazione della rete
- Marketplace AWS per l'infrastruttura di rete
- Amazon Virtual Private Cloud Connectivity Options Whitepaper
- Multiple data center HA network connectivity
- Connecting Networks with Overlapping IP Ranges
- What Is Amazon VPC?
- What is IPAM?

Video correlati:

- AWS re:Invent 2023 - Advanced VPC designs and new capabilities
- AWS re:Invent 2019: AWS Transit Gateway reference architectures for many VPCs
- AWS re:Invent 2023 - Ready for what's next? Designing networks for growth and flexibility
- AWS re:Invent 2021 - {New Launch} Manage your IP addresses at scale on AWS

Architettura del carico di lavoro

Questions

- REL 3. Come si progetta l'architettura del servizio di carico di lavoro?

- REL 4. Come si progettano le interazioni in un sistema distribuito per evitare errori?
- REL 5. Come si progettano le interazioni in un sistema distribuito per mitigare o affrontare gli errori?

REL 3. Come si progetta l'architettura del servizio di carico di lavoro?

Crea carichi di lavoro altamente scalabili e affidabili utilizzando un'architettura orientata ai servizi (SOA) o un'architettura di microservizi. L'architettura orientata ai servizi (SOA) è la pratica di rendere i componenti software riutilizzabili tramite interfacce di servizio. L'architettura dei microservizi va oltre, per rendere i componenti più piccoli e semplici.

Best practice

- REL03-BP01 Scegli come segmentare il tuo carico di lavoro
- REL03-BP02 Creazione di servizi focalizzati su domini e funzionalità aziendali specifici
- REL03-BP03 Fornire contratti di assistenza per API

REL03-BP01 Scegli come segmentare il tuo carico di lavoro

La segmentazione del carico di lavoro è importante nel determinare i requisiti di resilienza dell'applicazione. L'architettura monolitica va evitata, se possibile. Valuta invece con particolare attenzione quali componenti dell'applicazione possono essere suddivisi in microservizi. A seconda dei requisiti dell'applicazione, questa potrebbe finire per essere una combinazione di un'architettura orientata ai servizi () con microservizi, ove possibile. SOA I carichi di lavoro stateless sono più idonei all'implementazione come microservizi.

Risultato desiderato: i carichi di lavoro devono essere supportabili, scalabili e caratterizzati dal maggiore accoppiamento debole possibile.

Quando scegli come segmentare il carico di lavoro, trova il giusto compromesso tra i vantaggi e le complessità. Ciò che è giusto per un nuovo prodotto al primo lancio è diverso dai requisiti di un carico di lavoro creato per scalare le risorse. Durante la rifattorizzazione di un monolito esistente, dovrai considerare la capacità dell'applicazione di supportare la suddivisione in servizi stateless. La suddivisione dei servizi in elementi più piccoli consente a team ristretti e ben definiti di svilupparli e gestirli. Tuttavia, servizi di piccole dimensioni possono introdurre complessità, che includono un eventuale aumento della latenza, un debug più complesso e un maggiore carico operativo.

Anti-pattern comuni:

- Il microservizio Death Star rappresenta una situazione in cui i componenti atomici diventano così interdipendenti che un guasto verificatosi in un componente genera un guasto molto più grande, rendendo i componenti rigidi e fragili se considerati come monolito.

Vantaggi dell'adozione di questa best practice:

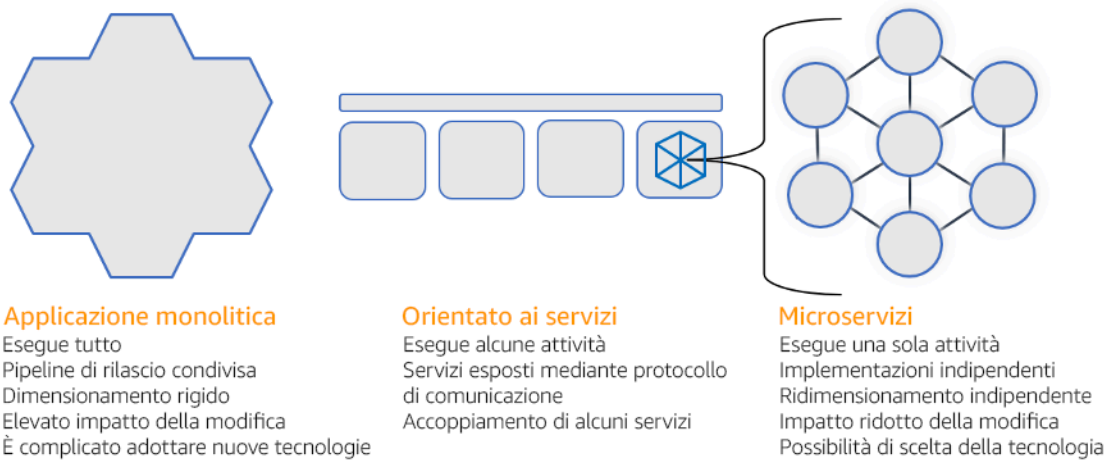
- Segmenti più specifici comportano maggiore agilità, flessibilità organizzativa e scalabilità.
- Riduzione dell'impatto derivante dall'interruzione dei servizi.
- I componenti dell'applicazione possono avere requisiti di disponibilità diversi, che a loro volta possono essere supportati da una segmentazione più atomica.
- Responsabilità ben definite per i team che supportano il carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Scegli il tipo di architettura in base al tipo di segmentazione del carico di lavoro. Scegliete un'SOAarchitettura a microservizi (o, in alcuni rari casi, un'architettura monolitica). Anche se scegli di iniziare con un'architettura monolitica, devi assicurarti che sia modulare e che alla fine possa evolversi verso i nostri microservizi man mano che il prodotto cresce con l'SOAadozione da parte degli utenti. SOAe i microservizi offrono rispettivamente una segmentazione più ridotta, che è preferibile in un'architettura moderna, scalabile e affidabile, ma ci sono dei compromessi da considerare, soprattutto quando si implementa un'architettura di microservizi.

Uno dei principali compromessi è che ora disponi di un'architettura di calcolo distribuita che può rendere più difficile il raggiungimento dei requisiti di latenza degli utenti ed è presente un'ulteriore complessità nel debug e nel tracciamento delle interazioni degli utenti. Puoi utilizzare AWS X-Ray per risolvere questo problema. Un altro effetto da considerare è l'aumento della complessità operativa man mano che aumenta il numero di applicazioni che gestisci, che richiede l'implementazione di più componenti di indipendenza.



Architettura monolitica, orientata ai servizi e di microservizi

Passaggi dell'implementazione

- Determina l'architettura più opportuna per rifattorizzare o creare l'applicazione. SOAe i microservizi offrono rispettivamente una segmentazione più piccola, preferibile come architettura moderna, scalabile e affidabile. SOApuò essere un buon compromesso per ottenere una segmentazione più piccola evitando al contempo alcune delle complessità dei microservizi. Per ulteriori dettagli, consulta Microservice Trade-Offs.
- Se il carico di lavoro è adatto e la tua organizzazione può supportarla, è consigliabile utilizzare un'architettura di microservizi per ottenere la massima agilità e affidabilità. Per ulteriori dettagli, consulta Implementazione dei microservizi su. AWS
- Valuta l'idea di attenerti al modello Strangler Fig per rifattorizzare un monolite in componenti più piccoli. Ciò comporta la sostituzione graduale di componenti applicativi specifici con nuove applicazioni e servizi. AWS Migration Hub Refactor Spaces funge da punto di partenza per procedere a rifattorizzare in modo incrementale. Per ulteriori dettagli, consulta Seamlessly migrate on-premises legacy workloads using a strangler pattern.
- L'implementazione dei microservizi può richiedere un meccanismo di rilevamento dei servizi per consentire a questi servizi distribuiti di comunicare tra loro. AWS App Meshpuò essere utilizzato con architetture orientate ai servizi per fornire l'individuazione e l'accesso affidabili ai servizi. AWS Cloud Mappuò essere utilizzato anche per l'individuazione dinamica dei servizi. DNS
- Se stai migrando da un monolite a, Amazon SOA MQ può aiutarti a colmare il divario come bus di servizio durante la riprogettazione delle applicazioni legacy nel cloud.

- Per i monoliti esistenti con un unico database condiviso, scegli come riorganizzare i dati in segmenti più piccoli. Questa riorganizzazione può avvenire per business unit, schema di accesso o struttura dei dati. A questo punto del processo di refactoring, dovresti scegliere di procedere con un tipo di database relazionale o non relazionale (No). SQL Per maggiori dettagli, consulta From to No. SQL SQL

Livello di impegno per il piano di implementazione: elevato

Risorse

Best practice correlate:

- REL03-BP02 Creazione di servizi focalizzati su domini e funzionalità aziendali specifici

Documenti correlati:

- Amazon API Gateway: configurazione di un REST API utilizzo di Open API
- Cosa si intende per SOA (architettura orientata ai servizi)?
- Bounded Context (un modello centrale in Domain-Driven Design)
- Implementazione di microservizi su AWS
- Microservice Trade-Offs
- Microservices - a definition of this new architectural term
- Microservizi attivi AWS
- Che cos'è AWS App Mesh?

Esempi correlati:

- Iterative App Modernization Workshop

Video correlati:

- Offrire l'eccellenza con i microservizi attivi AWS

REL03-BP02 Creazione di servizi focalizzati su domini e funzionalità aziendali specifici

L'architettura orientata ai servizi (SOA) definisce servizi con funzioni ben delineate determinate dalle esigenze aziendali. I microservizi utilizzano modelli di dominio e contesto delimitato per tracciare i limiti dei servizi lungo i confini del contesto aziendale. Concentrarsi sui domini e sulle funzionalità aziendali aiuta i team a definire requisiti di affidabilità indipendenti per i propri servizi. I contesti delimitati isolano e incapsulano la logica aziendale, consentendo ai team di ragionare meglio su come gestire gli errori.

Risultato desiderato: ingegneri e parti interessate aziendali definiscono congiuntamente contesti delimitati e li utilizzano per progettare sistemi come servizi che soddisfano funzioni aziendali specifiche. Questi team utilizzano pratiche consolidate come l'event storming per definire i requisiti. Le nuove applicazioni sono concepite come servizi con confini ben definiti e con accoppiamento debole. Avviene la scomposizione dei monoliti esistenti in contesti delimitati e i progetti di sistema migrano verso architetture SOA o microservizi. In caso di rifattorizzazione dei monoliti, vengono applicati approcci consolidati come contesti a bolle e schemi di decomposizione dei monoliti.

I servizi orientati al dominio vengono eseguiti come uno o più processi che non condividono lo stato. Rispondono in modo indipendente alle fluttuazioni della domanda e gestiscono gli scenari di errore alla luce dei requisiti specifici del dominio.

Anti-pattern comuni:

- I team sono formati su domini tecnici specifici come UI e UX, middleware (software intermediario) o database anziché su domini aziendali specifici.
- Le applicazioni coprono le responsabilità di dominio. I servizi che coprono contesti delimitati possono essere più difficili da gestire, richiedere maggiori sforzi di test ed esigere la partecipazione di più team di dominio agli aggiornamenti software.
- Le dipendenze a livello di dominio, come le librerie di entità di dominio, sono condivise tra i servizi, in modo che le modifiche per il dominio di un servizio richiedano modifiche ad altri domini dei servizi.
- I contratti di servizio e la logica aziendale non esprimono le entità in un linguaggio di dominio comune e coerente, con il risultato di livelli di traduzione che complicano i sistemi e aumentano le attività di debug.

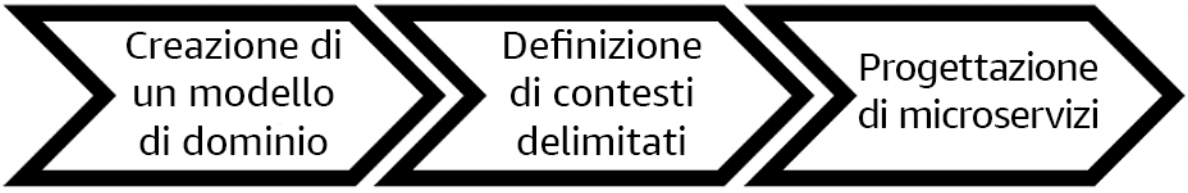
Vantaggi dell'adozione di questa best practice: le applicazioni sono progettate come servizi indipendenti limitati da domini aziendali e utilizzano un linguaggio aziendale comune. I servizi sono

testabili e implementabili in modo indipendente. I servizi soddisfano i requisiti di resilienza specifici del dominio implementato.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

La progettazione basata sul dominio (DDD) costituisce l'approccio fondamentale alla progettazione e alla creazione di software attorno ai domini aziendali. È utile utilizzare un framework esistente quando si creano servizi incentrati sui domini aziendali. Quando si utilizzano applicazioni monolitiche esistenti, è possibile sfruttare i modelli di decomposizione che forniscono tecniche consolidate per modernizzare le applicazioni in servizi.



Progettazione basata sul dominio

Passaggi dell'implementazione

- I team possono organizzare workshop di event storming per identificare rapidamente eventi, comandi, aggregati e domini in un formato leggero simile a quello delle note adesive.
- Una volta create le entità e le funzioni di dominio in un contesto di dominio, puoi suddividere il dominio in servizi mediante il contesto delimitato, che raggruppa entità con funzionalità e attributi simili. Con il modello diviso in contesti, emerge un modello su come delimitare i microservizi.
  - Ad esempio, le entità del sito Web Amazon.com possono includere elementi quali pacchetti, distribuzione, pianificazione, prezzo, sconto e valuta.
  - Il pacchetto, la distribuzione e la pianificazione sono raggruppati nel contesto di spedizione, mentre il prezzo, lo sconto e la valuta sono raggruppati nel contesto dei prezzi.
- La scomposizione dei monoliti in microservizi delinea i modelli per rifattorizzare i microservizi. L'utilizzo di modelli per la decomposizione in base a capacità aziendale, sottodominio o transazione si allinea bene agli approcci basati sul dominio.
- Tecniche di strategia come il contesto a bolle consentono di introdurre la decisione basata sul dominio (DDD) in applicazioni esistenti o legacy senza riscritture anticipate e impegni completi nei confronti di DDD. In un approccio basato sul contesto delle bolle, si crea un contesto ristretto e

delimitato mediante un livello di mappatura e coordinamento dei servizi o il livello anticorruzione, che protegge il modello di dominio appena definito dalle influenze esterne.

Dopo aver eseguito l'analisi del dominio e definito le entità e i contratti di servizio, i team possono utilizzare i servizi AWS per implementare la progettazione basata sul dominio come servizi basati sul cloud.

- Inizia a sviluppare definendo test che applichino le regole aziendali del tuo dominio. Lo sviluppo basato sui test (TDD) e lo sviluppo basato sul comportamento (BDD) aiutano i team a focalizzare i servizi sulla risoluzione dei problemi aziendali.
- Seleziona i servizi AWS ideali per i requisiti del tuo dominio aziendale e l'architettura dei microservizi:
  - AWS Serverless consente al team di concentrarsi su una logica di dominio specifica anziché sulla gestione di server e infrastrutture.
  - I container in AWS semplificano la gestione della tua infrastruttura, in modo da poterti concentrare sui requisiti del tuo dominio.
  - I database dedicati ti aiutano ad adattare i requisiti del tuo dominio al tipo di database più idoneo.
- La creazione di architetture esagonali in AWS delinea un framework per integrare la logica aziendale nei servizi che funzionano a ritroso da un dominio aziendale per soddisfare i requisiti funzionali e, quindi, per collegare adattatori di integrazione. I modelli che separano i dettagli dell'interfaccia dalla logica aziendale con i servizi AWS aiutano i team a concentrarsi sulla funzionalità del dominio e a migliorare la qualità del software.

Risorse

Best practice correlate:

- REL03-BP01 Scegli come segmentare il tuo carico di lavoro
- REL03-BP03 Fornire contratti di assistenza per API

Documenti correlati:

- Microservizi AWS
- Implementazione di microservizi in AWS
- How to break a Monolith into Microservices

- Getting Started with DDD when Surrounded by Legacy Systems
- Domain-Driven Design: Tackling Complexity in the Heart of Software
- Building hexagonal architectures on AWS
- Decomposing monoliths into microservices
- Event Storming
- Messages Between Bounded Contexts
- Microservices
- Sviluppo basato su test
- Sviluppo basato sul comportamento

Esempi correlati:

- Progettazione di microservizi cloud-native (nativi del cloud) su AWS (da DDD/EventStormingWorkshop)

Strumenti correlati:

- Database Cloud AWS
- Serverless in AWS
- Container in AWS

REL03-BP03 Fornire contratti di assistenza per API

I contratti di assistenza sono accordi documentati tra API produttori e consumatori definiti in una definizione leggibile da una macchinaAPI. Una strategia di controllo delle versioni contrattuali consente ai consumatori di continuare a utilizzare le applicazioni esistenti API e di migrare le proprie applicazioni a una versione più recente quando sono pronte. API L'implementazione da parte del produttore può avvenire in qualsiasi momento, purché il processo sia conforme al contratto. I team di assistenza possono utilizzare lo stack tecnologico di loro scelta per soddisfare il contratto. API

Risultato desiderato: le applicazioni create con architetture orientate ai servizi o ai microservizi sono in grado di funzionare in modo indipendente pur avendo una dipendenza di runtime integrata. Le modifiche apportate a un API consumatore o a un produttore non interrompono la stabilità dell'intero sistema quando entrambe le parti seguono un contratto comune. API I componenti che comunicano tramite servizio APIs possono eseguire rilasci funzionali indipendenti, aggiornamenti alle dipendenze

di runtime o eseguire il failover su un sito di disaster recovery (DR) con un impatto reciproco minimo o nullo. Inoltre, i servizi discreti sono in grado di scalare in modo indipendente assorbendo la richiesta di risorse senza che gli altri servizi debbano ridurre orizzontalmente di conseguenza.

Anti-pattern comuni:

- Creazione di servizi APIs senza schemi fortemente tipizzati. Ciò comporta APIs che non può essere utilizzato per generare API associazioni e payload che non possono essere convalidati programmaticamente.
- Non adottare una strategia di controllo delle versioni, che costringerebbe gli API utenti ad aggiornare e rilasciare o fallire quando i contratti di assistenza si evolvono.
- Messaggi di errore che divulgano dettagli sull'implementazione del servizio sottostante anziché descrivere errori di integrazione nel contesto e nel linguaggio del dominio.
- Non utilizzare API contratti per sviluppare casi di test e API implementazioni fittizie per consentire test indipendenti dei componenti del servizio.

Vantaggi derivanti dall'adozione di questa best practice: i sistemi distribuiti composti da componenti che comunicano tramite contratti di API assistenza possono migliorare l'affidabilità. Gli sviluppatori possono individuare potenziali problemi nelle prime fasi del processo di sviluppo grazie al controllo del tipo durante la compilazione per verificare che le richieste e le risposte siano conformi al API contratto e che i campi obbligatori siano presenti. APIi contratti forniscono una chiara interfaccia di autodocumentazione APIs e forniscono una migliore interoperabilità tra diversi sistemi e linguaggi di programmazione.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Dopo aver identificato i domini aziendali e determinato la segmentazione del carico di lavoro, puoi sviluppare il tuo servizio. APIs Innanzitutto, definisci contratti di assistenza leggibili automaticamente e poi implementa una strategia di controllo delle APIs versioni. API Quando sei pronto per integrare i servizi tramite protocolli comuni come REST GraphQL o eventi asincroni, puoi incorporare AWS servizi nella tua architettura per integrare i componenti con contratti fortemente tipizzati. API

AWS APIservizi per contratti di assistenza

Incorpora AWS servizi tra cui Amazon API Gateway e Amazon EventBridge nella tua architettura per utilizzare i contratti di API servizio nella tua applicazione. AWS AppSync Amazon API Gateway

ti aiuta a integrarti con AWS servizi nativi diretti e altri servizi Web. APIGateway supporta le APIspecifiche e il controllo delle versioni Open. AWS AppSync è un endpoint GraphQL gestito che puoi configurare definendo uno schema GraphQL per definire un'interfaccia di servizio per query, mutazioni e sottoscrizioni. Amazon EventBridge utilizza schemi di eventi per definire eventi e generare associazioni di codice per i tuoi eventi.

Passaggi dell'implementazione

- Innanzitutto, definisci un contratto per il tuo. API Un contratto esprimerà le capacità di un e API definirà oggetti di dati e campi fortemente tipizzati per l'APIinput e l'output.
- Quando esegui la configurazione APIs in API Gateway, puoi importare ed esportare API le specifiche aperte per i tuoi endpoint.
  - L'importazione di una API definizione aperta semplifica la creazione della propria definizione API e può essere integrata con l' AWS infrastruttura come strumenti di codice come la e. AWS Serverless Application ModelAWS Cloud Development Kit (AWS CDK)
  - L'esportazione di una API definizione semplifica l'integrazione con gli strumenti di API test e fornisce ai consumatori di servizi una specifica di integrazione.
- Puoi definire e gestire GraphQL APIs AWS AppSync definendo un file di schema GraphQL per generare l'interfaccia del contratto e semplificare l'interazione con REST modelli complessi, più tabelle di database o servizi legacy.
- AWS Amplifyi progetti integrati AWS AppSync generano file di JavaScript query fortemente tipizzati da utilizzare nella tua applicazione e una libreria client AWS AppSync GraphQL per le tabelle Amazon DynamoDB.
- Quando utilizzi gli eventi di servizio di Amazon EventBridge, gli eventi aderiscono a schemi già esistenti nel registro degli schemi o definiti con Open API Spec. Con uno schema definito nel registro, puoi anche generare associazioni client dal contratto dello schema per integrare il codice con gli eventi.
- Estendere o modificare il tuo. API L'estensione di un API è un'opzione più semplice quando si aggiungono campi che possono essere configurati con campi opzionali o valori predefiniti per i campi obbligatori.
  - JSONi contratti basati su protocolli come REST GraphQL possono essere adatti per l'estensione del contratto.
  - XMLi contratti basati su protocolli, ad esempio, SOAP dovrebbero essere testati con i consumatori di servizi per determinare la fattibilità dell'estensione del contratto.

- Quando si effettua il versionamento di unAPI, è consigliabile implementare il controllo delle versioni proxy, in cui viene utilizzata una facciata per supportare le versioni in modo che la logica possa essere mantenuta in un'unica base di codice.
  - Con API Gateway puoi utilizzare le mappature di richiesta e risposta per semplificare l'assorbimento delle modifiche contrattuali, stabilendo una facciata per fornire valori predefiniti per nuovi campi o per eliminare i campi rimossi da una richiesta o risposta. Con questo approccio, il servizio sottostante può avere un'unica base di codice.

Risorse

Best practice correlate:

- REL03-BP01 Scegli come segmentare il tuo carico di lavoro
- REL03-BP02 Creazione di servizi focalizzati su domini e funzionalità aziendali specifici
- REL04-BP02 Implementare dipendenze liberamente accoppiate
- REL05-BP03 Controlla e limita le chiamate di nuovo tentativo
- REL05-BP05 Imposta i timeout dei client

Documenti correlati:

- Che cos'è una API (interfaccia di programmazione delle applicazioni)?
- Implementazione di microservizi su AWS
- Microservice Trade-Offs
- Microservices - a definition of this new architectural term
- Microservizi attivi AWS
- Utilizzo delle estensioni API Gateway to Open API
- Apri API - Specificazione
- GraphQL: schemi e tipi
- Associazioni di EventBridge codice Amazon

Esempi correlati:

- Amazon API Gateway: configurazione di un REST API utilizzo di Open API
- Da Amazon API Gateway all'applicazione Amazon CRUD DynamoDB tramite Open API

- Modelli di integrazione delle applicazioni moderni nell'era senza server: API Gateway Service Integration
- Implementazione del controllo delle versioni API Gateway basato su header con Amazon CloudFront
- AWS AppSync: Building a client application

Video correlati:

- Utilizzo di Open API in AWS SAM per gestire Gateway API

Strumenti correlati:

- Amazon API Gateway
- AWS AppSync
- Amazon EventBridge

REL 4. Come si progettano le interazioni in un sistema distribuito per evitare errori?

I sistemi distribuiti si basano sulle reti di comunicazione per interconnettere i componenti, ad esempio server o servizi. Il carico di lavoro deve funzionare in modo affidabile nonostante la perdita o la latenza dei dati su queste reti. I componenti del sistema distribuito devono funzionare in modo da non influire negativamente su altri componenti o sul carico di lavoro. Queste best practice consentono di prevenire gli errori e migliorare il tempo medio tra guasti (MTBF).

Best practice

- REL04-BP01 Identificazione del tipo di sistema distribuito da cui si dipende
- REL04-BP02 Implementare dipendenze liberamente accoppiate
- REL04-BP03 Fai un lavoro costante
- REL04-BP04 Rendere idempotenti le operazioni di mutazione

REL04-BP01 Identificazione del tipo di sistema distribuito da cui si dipende

I sistemi distribuiti possono essere sincroni, asincroni o batch. I sistemi sincroni devono elaborare le richieste il più rapidamente possibile e comunicare tra loro effettuando chiamate di richiesta e risposta sincrone utilizzando i protocolli HTTP/S, REST o RPC (Remote Procedure Call). I sistemi

asincroni comunicano tra loro scambiando i dati in modo asincrono tramite un servizio intermediario senza associare singoli sistemi. I sistemi batch ricevono un grande volume di dati di input, eseguono i processi di dati automatizzati senza intervento umano e generano i dati di output.

Risultato desiderato: progettazione di un carico di lavoro in grado di interagire in modo efficace con dipendenze sincrone, asincrone e batch.

Anti-pattern comuni:

- Il carico di lavoro attende a tempo indeterminato una risposta dalle dipendenze, con eventuale timeout del client del carico di lavoro, senza informazioni sulla ricezione della richiesta.
- Il carico di lavoro utilizza una catena di sistemi dipendenti che effettuano chiamate reciproche in modo sincrono. A tal fine, ogni sistema deve essere disponibile ed elaborare correttamente la richiesta prima che l'intera catena possa essere completata, con conseguenti comportamenti e disponibilità complessiva potenzialmente fragili.
- Il carico di lavoro comunica con le dipendenze in modo asincrono e si basa sul concetto di distribuzione garantita dei messaggi esattamente una volta, quando spesso è ancora possibile ricevere messaggi duplicati.
- Il carico di lavoro non utilizza strumenti di pianificazione batch adeguati e consente l'esecuzione simultanea dello stesso processo batch.

Vantaggi dell'adozione di questa best practice: non è insolito che un determinato carico di lavoro implementi uno o più stili di comunicazione tra sincroni, asincroni e batch. Questa best practice consente di identificare i diversi compromessi associati a ogni stile di comunicazione per rendere il carico di lavoro in grado di tollerare interruzioni in tutte le sue dipendenze.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Le sezioni seguenti contengono le linee guida per l'implementazione generali e specifiche di ogni tipo di dipendenza.

Informazioni generali

- Assicurati che gli obiettivi del livello di servizio (SLO) in termini di prestazioni e affidabilità offerti dalle dipendenze soddisfino i requisiti di prestazioni e affidabilità del tuo carico di lavoro.
- Utilizza i servizi di osservabilità AWS per monitorare i tempi di risposta e i tassi di errore così da verificare che la tua dipendenza fornisca un servizio ai livelli richiesti dal carico di lavoro.

- Individua le potenziali sfide che il carico di lavoro può affrontare quando comunica con le dipendenze. I sistemi distribuiti presentano un'ampia gamma di sfide in grado di far aumentare complessità dell'architettura, carico operativo e costi. Le sfide più comuni includono latenza, interruzioni della rete, perdita dei dati, scalabilità e ritardo nella replica dei dati.
- Implementa una gestione e una creazione di log affidabili degli errori per risolvere i problemi quando si verificano quelli legati alle dipendenze.

Dipendenza sincrona

Nelle comunicazioni sincrone, il carico di lavoro invia una richiesta alla dipendenza e blocca l'operazione in attesa della risposta. Quando la dipendenza riceve la richiesta, cerca di gestirla il prima possibile e invia una risposta al carico di lavoro. Una sfida significativa con la comunicazione sincrona è rappresentata dall'accoppiamento temporale, che richiede che il carico di lavoro e le sue dipendenze siano disponibili nello stesso momento. Quando il carico di lavoro deve comunicare in modo sincrono con le dipendenze, valuta le seguenti linee guida:

- Il carico di lavoro non deve fare affidamento su più dipendenze sincrone per eseguire una singola funzione. Questa catena di dipendenze aumenta la fragilità complessiva perché tutte le dipendenze nel percorso devono essere disponibili affinché la richiesta venga completata correttamente.
- Quando una dipendenza non è integra o non è disponibile, applica le strategie di gestione degli errori e riprova. Evita di usare un comportamento bimodale. Il comportamento bimodale si verifica quando il carico di lavoro presenta un comportamento diverso in modalità normale e in modalità di guasto. Per ulteriori dettagli sul comportamento bimodale, consulta REL11-BP05 Utilizzo della stabilità statica per evitare un comportamento bimodale.
- Tieni presente che anticipare l'errore (fail fast) è meglio che far aspettare il carico di lavoro. Ad esempio, la guida per gli sviluppatori AWS Lambda illustra come gestire tentativi ed errori nel richiamare le funzioni Lambda.
- Imposta i timeout per le chiamate delle dipendenze da parte del carico di lavoro. Questa tecnica evita di aspettare troppo a lungo o all'infinito una risposta. Per un'utile discussione su questo argomento, consulta Tuning AWS Java SDK HTTP request settings for latency-aware Amazon DynamoDB applications.
- Riduci al minimo il numero di chiamate effettuate dal carico di lavoro alla dipendenza per soddisfare una singola richiesta. Le lunghe chiamate aumentano l'associazione e la latenza.

Dipendenza sincrona

Per disaccoppiare temporaneamente il carico di lavoro dalla dipendenza, è necessario che comunichino in modo asincrono. Con l'approccio asincrono, il carico di lavoro può continuare qualsiasi altra elaborazione senza dover attendere che la dipendenza o la catena di dipendenze invii la risposta.

Quando il carico di lavoro deve comunicare in modo asincrono con la dipendenza, tieni conto delle seguenti indicazioni:

- Determina in base al caso d'uso e ai requisiti se utilizzare la messaggistica o lo streaming di eventi. La messaggistica consente al carico di lavoro di comunicare con la relativa dipendenza, inviando e ricevendo messaggi tramite un broker di messaggi. Lo streaming di eventi consente al carico di lavoro e alla relativa dipendenza di utilizzare un servizio di streaming per la pubblicazione e l'abbonamento a eventi, forniti come flussi continui di dati, da elaborare il prima possibile.
- La messaggistica e lo streaming di eventi gestiscono i messaggi in modo diverso, quindi devi stabilire i compromessi in base a:
  - Priorità dei messaggi: i broker di messaggi sono in grado di elaborare messaggi ad alta priorità prima di quelli normali. Nello streaming di eventi, tutti i messaggi presentano la stessa priorità.
  - Consumo di messaggi: i broker di messaggi garantiscono la ricezione del messaggio da parte dei consumatori. Gli utenti che utilizzano lo streaming di eventi devono tenere traccia dell'ultimo messaggio letto.
  - Ordinamento dei messaggi: utilizzando la messaggistica, la ricezione dei messaggi nell'ordine esatto di invio non è garantita, salvo in caso di un approccio first-in-first-out (FIFO). Lo streaming di eventi mantiene sempre l'ordine in cui i dati sono stati prodotti.
  - Eliminazione dei messaggi: con la messaggistica, il consumatore deve eliminare il messaggio dopo la relativa elaborazione. Il servizio di streaming di eventi aggiunge il messaggio a un flusso e lo conserva fino alla scadenza del periodo di conservazione del messaggio. Questa policy di eliminazione rende lo streaming di eventi adatto alla riproduzione dei messaggi.
- Definisci in che modo il carico di lavoro riconosce il completamento del lavoro della dipendenza. Ad esempio, quando il carico di lavoro procede a richiamare una funzione Lambda in modo asincrono, Lambda inserisce la richiesta in una coda e restituisce una risposta di esito positivo senza ulteriori informazioni. Al termine dell'elaborazione, la funzione Lambda può inviare il risultato a una destinazione, configurabile in base all'esito positivo o negativo.
- Crea il tuo carico di lavoro per gestire i messaggi duplicati utilizzando l'idempotenza. Con l'idempotenza i risultati del carico di lavoro non cambiano anche se il carico di lavoro viene generato più volte per lo stesso messaggio. È importante sottolineare che i servizi di messaggistica

o streaming recapiteranno di nuovo un messaggio in caso di errore di rete o mancata ricezione della conferma.

- Se il carico di lavoro non riceve una risposta dalla dipendenza, deve inviare nuovamente la richiesta. Valuta la possibilità di limitare il numero di tentativi per preservare la CPU, la memoria e le risorse di rete del carico di lavoro al fine di gestire le altre richieste. La documentazione AWS Lambda illustra come gestire gli errori di invocazione asincrona.
- Utilizza gli strumenti di osservabilità, debug e monitoraggio adeguati per gestire e usare la comunicazione asincrona del carico di lavoro con le relative dipendenze. Amazon CloudWatch ti consente di monitorare i servizi di messaggistica e streaming di eventi. Puoi anche dotare di strumenti il tuo carico di lavoro con AWS X-Ray per ottenere informazioni utili rapidamente per la risoluzione dei problemi.

Dipendenza dal batch

I sistemi batch acquisiscono i dati di input, avviano una serie di processi per elaborarli e producono i dati di output, senza intervento manuale. A seconda delle dimensioni dei dati, i processi possono durare da minuti a diversi giorni in alcuni casi. Quando il carico di lavoro comunica con la dipendenza batch, tieni conto delle seguenti indicazioni:

- Definisci la finestra temporale in cui il carico di lavoro deve eseguire il processo batch. Puoi impostare un modello di ricorrenza per il carico di lavoro per richiamare il sistema batch, ad esempio ogni ora o alla fine di ogni mese.
- Determina la posizione dei dati di input e di output elaborati. Scegli un servizio di archiviazione, come Amazon Simple Storage Service (Amazon S3), Amazon Elastic File System (Amazon EFS) e Amazon FSx per Lustre, in modo che il tuo carico di lavoro possa leggere e scrivere file su larga scala.
- Se il tuo carico di lavoro deve richiamare più processi batch, puoi sfruttare AWS Step Functions per semplificare l'orchestrazione dei processi batch eseguiti in AWS o on-premises. Questo progetto di esempio mostra l'orchestrazione di processi batch utilizzando Step Functions, AWS Batch e Lambda.
- Monitora i processi batch per individuare eventuali anomalie, ad esempio un processo che richiede più tempo del dovuto per essere completato. Puoi utilizzare strumenti come CloudWatch Container Insights per monitorare ambienti e processi AWS Batch. In tal caso, il carico di lavoro interrompe l'inizio del processo successivo e comunica l'eccezione al team competente.

Risorse

Documenti correlati:

- Cloud AWS Operations: monitoraggio e osservabilità
- Amazon Builders' Library: difficoltà dei sistemi distribuiti
- REL11-BP05 Utilizzo della stabilità statica per evitare un comportamento bimodale
- AWS Lambda Developer Guide: Error handling and automatic retries in AWS Lambda
- Tuning AWS Java SDK HTTP request settings for latency-aware Amazon DynamoDB applications
- Messaggi AWS
- Cosa sono i flussi di dati?
- AWS Lambda Developer Guide: Asynchronous invocation
- Amazon Simple Queue Service FAQ: FIFO queues
- Amazon Kinesis Data Streams Developer Guide: Handling Duplicate Records
- Amazon Simple Queue Service Developer Guide: Available CloudWatch metrics for Amazon SQS
- Amazon Kinesis Data Streams Developer Guide: Monitoring the Amazon Kinesis Data Streams Service with Amazon CloudWatch
- AWS X-Ray Developer Guide: AWS X-Ray concepts
- Esempi AWS su GitHub: AWS Step functions Complex Orchestrator App
- Guida per l'utente AWS Batch: AWS Batch CloudWatch Container Insights

Video correlati:

- AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS (COP310)

Strumenti correlati:

- Amazon CloudWatch
- Amazon CloudWatch Logs
- AWS X-Ray
- Amazon Simple Storage Service (Amazon S3)
- Amazon Elastic File System (Amazon EFS)

- Amazon FSx per Lustre
- AWS Step Functions
- AWS Batch

REL04-BP02 Implementare dipendenze liberamente accoppiate

Le dipendenze come sistemi di accodamento, sistemi di streaming, flussi di lavoro e bilanciatori del carico sono con accoppiamento debole. L'accoppiamento debole aiuta a isolare il comportamento di un componente dagli altri componenti che dipendono da esso, aumentando la resilienza e l'agilità.

Il disaccoppiamento delle dipendenze, come i sistemi di coda, quelli di streaming e i flussi di lavoro, favorisce la riduzione al minimo dell'impatto sul sistema di modifiche o guasti. Tale separazione isola il comportamento di un componente dall'impatto sugli altri dipendenti dallo stesso, migliorando resilienza e agilità.

Nei sistemi con accoppiamento stretto, le modifiche a un componente possono richiedere modifiche agli altri componenti basati su di esso, con conseguente riduzione delle prestazioni di tutti i componenti. L'accoppiamento debole interrompe questa dipendenza, in modo che i componenti dipendenti debbano conoscere solo l'interfaccia con versione e pubblicata. L'implementazione di un accoppiamento debole tra dipendenze isola un errore all'interno di una dipendenza affinché non influenzi l'altra.

L'accoppiamento debole consente di modificare il codice o aggiungere funzionalità a un componente riducendo al minimo il rischio per gli altri componenti che dipendono da esso. Garantisce inoltre una resilienza granulare a livello di componente in cui è possibile aumentare orizzontalmente o persino modificare l'implementazione sottostante della dipendenza.

Per migliorare ulteriormente la resilienza tramite accoppiamento debole, rendi le interazioni dei componenti asincrone laddove possibile. Questo modello è idoneo a qualsiasi interazione che non richieda una risposta immediata e laddove la conferma della registrazione di una richiesta sia sufficiente. Include un componente che genera eventi e un altro che li utilizza. I due componenti non si integrano tramite un' point-to-pointinterazione diretta, ma di solito attraverso un livello di storage intermedio durevole, come una SQS coda Amazon, una piattaforma di dati di streaming come Amazon Kinesis o. AWS Step Functions

Figura 4: dipendenze come sistemi di accodamento e bilanciatori del carico con accoppiamento debole

Amazon mette in SQS coda e AWS Step Functions sono solo due modi per aggiungere uno strato intermedio per l'accoppiamento libero. Le architetture basate sugli eventi possono anche essere create utilizzando Cloud AWS Amazon EventBridge, che può astrarre i clienti (produttori di eventi) dai servizi su cui fanno affidamento (consumatori di eventi). Amazon Simple Notification Service (AmazonSNS) è una soluzione efficace quando è necessaria una messaggistica basata su push ad alta velocità. many-to-many Utilizzando SNS gli argomenti di Amazon, i tuoi sistemi di pubblicazione possono inviare messaggi a un gran numero di endpoint di abbonati per l'elaborazione parallela.

Mentre le code offrono diversi vantaggi, nella maggior parte dei sistemi hard real-time, le richieste più vecchie di una soglia temporale (spesso secondi) dovrebbero essere considerate obsolete (il client ha abbandonato e non è più in attesa di una risposta) e non elaborate. In questo modo, è possibile elaborare invece le richieste più recenti (e probabilmente ancora valide).

Risultato desiderato: riduzione al minimo l'area della superficie in caso di guasto a livello di componente, supportando così diagnostica e risoluzione dei problemi, grazie all'implementazione di dipendenze con accoppiamento debole. Inoltre, semplifica i cicli di sviluppo, consentendo ai team di implementare le modifiche a livello modulare senza pregiudicare le prestazioni di altri componenti che dipendono da esso. Questo approccio offre la possibilità di aumentare orizzontalmente a livello di componente in base al fabbisogno di risorse, nonché di utilizzare un componente che contribuisce alla competitività in termini di costi.

Anti-pattern comuni:

- Implementazione di un carico di lavoro monolitico.
- Richiamo diretto APIs tra livelli di carico di lavoro senza possibilità di failover o elaborazione asincrona della richiesta.
- Accoppiamento stretto utilizzando dati condivisi. I sistemi con accoppiamento debole dovrebbero evitare di condividere i dati tramite database condivisi o altre forme di archiviazione di dati con accoppiamento stretto, che possono reintrodurre l'accoppiamento stretto e compromettere la scalabilità.
- Ignorare la contropressione. Il carico di lavoro dovrebbe essere in grado di rallentare o arrestare i dati in arrivo quando un componente non è in grado di elaborarli alla stessa velocità.

Vantaggi dell'adozione di questa best practice: l'accoppiamento debole aiuta a isolare il comportamento di un componente dagli altri componenti che dipendono da esso, aumentando la resilienza e l'agilità. L'errore in un componente è isolato dagli altri.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Implementazione di dipendenze con accoppiamento debole Esistono varie soluzioni che consentono di creare applicazioni con accoppiamento debole. Questi includono servizi per l'implementazione di code completamente gestite, flussi di lavoro automatizzati, la reazione agli eventi e, APIs tra gli altri, che possono aiutare a isolare il comportamento dei componenti dagli altri componenti e, di conseguenza, aumentare la resilienza e l'agilità.

- Crea architetture basate sugli eventi: EventBridgeAmazon ti aiuta a creare architetture basate sugli eventi liberamente accoppiate e distribuite.
- Implementazione di code in sistemi distribuiti: puoi utilizzare Amazon Simple Queue Service SQS (Amazon) per integrare e disaccoppiare sistemi distribuiti.
- Containerizza i componenti come microservizi: i microservizi consentono ai team di creare applicazioni composte da piccoli componenti indipendenti che comunicano in modo ben definito. APIs Amazon Elastic Container Service (AmazonECS) e Amazon Elastic Kubernetes Service (EKSAmazon) possono aiutarti a iniziare a usare i container più velocemente.
- Gestisci i flussi di lavoro con Step Functions: Step Functions ti aiuta a coordinare più AWS servizi in flussi di lavoro flessibili.
- Sfrutta le architetture di messaggistica publish-subscribe (pub/sub): Amazon Simple Notification Service (Amazon SNS) fornisce il recapito dei messaggi dagli editori agli abbonati (noti anche come produttori e consumatori).

Passaggi dell'implementazione

- I componenti in un'architettura basata su eventi vengono avviati dagli eventi. Gli eventi sono azioni che si verificano in un sistema, ad esempio un utente che aggiunge un articolo a un carrello. Quando un'azione ha successo, viene generato un evento che attiva il successivo componente del sistema.
  - Creazione di applicazioni basate sugli eventi con Amazon EventBridge
  - AWS re:Invent 2022 - Progettazione di integrazioni basate sugli eventi con Amazon EventBridge
- I sistemi di messaggistica distribuiti sono composti da tre parti principali che devono essere implementate per un'architettura basata su code. Includono componenti del sistema distribuito, la coda utilizzata per il disaccoppiamento (distribuita sui SQS server Amazon) e i messaggi in coda. Un sistema tipico prevede produttori che inviano il messaggio alla coda e il consumatore che riceve il messaggio dalla coda. La coda archivia i messaggi su più SQS server Amazon per motivi di ridondanza.

  - SQSArchitettura Amazon di base
  - Send Messages Between Distributed Applications with Amazon Simple Queue Service
- I microservizi, se ben utilizzati, migliorano la manutenibilità e aumentano la scalabilità, poiché i componenti ad accoppiamento debole sono gestiti da team indipendenti. Consentono inoltre l'isolamento dei comportamenti in un unico componente in caso di modifiche.
  - Implementazione di microservizi su AWS
  - Let's Architect! Architecting microservices with containers
- Con AWS Step Functions puoi creare applicazioni distribuite, automatizzare i processi, orchestrare microservizi, tra le altre cose. L'orchestrazione di più componenti in un flusso di lavoro automatizzato consente di disaccoppiare le dipendenze nell'applicazione.
  - Crea un flusso di lavoro serverless con e AWS Step FunctionsAWS Lambda
  - Guida introduttiva con AWS Step Functions

Risorse

Documenti correlati:

- AmazonEC2: garantire l'idempotenza
- The Amazon Builders' Library: difficoltà dei sistemi distribuiti
- The Amazon Builders' Library: Reliability, constant work, and a good cup of coffee
- Che cos'è Amazon EventBridge?
- What Is Amazon Simple Queue Service?
- Break up with your monolith
- Orchestra i microservizi basati sulle code con Amazon AWS Step Functions SQS
- SQSArchitettura Amazon di base
- Queue-Based Architecture

Video correlati:

- AWS New York Summit 2019: introduzione alle architetture basate sugli eventi e ad Amazon (05) EventBridge MAD2
- AWS re:Invent 2018: Close Loops and Opening Minds: Come prendere il controllo di sistemi, grandi e piccoli ARC337 (include accoppiamento libero, lavoro costante, stabilità statica)

- AWS re:Invent 2019: Passaggio ad architetture basate sugli eventi (08) SVS3
- AWS re:Invent 2019: applicazioni scalabili senza server basate su eventi con Amazon e Lambda SQS
- AWS re:Invent 2022 - Progettazione di integrazioni basate sugli eventi con Amazon EventBridge
- AWS re:Invent 2017: Approfondimento e best practice su Elastic Load Balancing

REL04-BP03 Fai un lavoro costante

I sistemi possono presentare guasti quando si verificano modifiche rapide e di grandi dimensioni nel carico. Ad esempio, se il carico di lavoro effettua un controllo dell'integrità di migliaia di server deve inviare ogni volta lo stesso payload delle dimensioni (uno snapshot completo dello stato corrente). Indipendentemente dal fatto che non ci siano server guasti, o che lo siano tutti, il sistema di controllo dell'integrità esegue un lavoro costante con modifiche rapide e di piccole dimensioni.

Ad esempio, se il sistema di controllo dell'integrità monitora 100.000 server, il carico su di esso è nominale al di sotto del tasso di errore normalmente basso del server. Tuttavia, se un evento importante rendesse la metà di questi server non integra, il sistema di controllo dell'integrità sarebbe sovraccarico nel tentativo di aggiornare i sistemi di notifica e comunicare lo stato con i client. Pertanto, il sistema di controllo dell'integrità dovrebbe inviare ogni volta lo snapshot completo dello stato attuale. 100.000 stati di integrità del server, ciascuno rappresentato da un bit, equivarrebbero a un payload di soli 12,5 KB. Indipendentemente dal fatto che non ci siano server guasti, o che lo siano tutti, il sistema di controllo dell'integrità esegue un lavoro costante e le modifiche rapide e di grandi dimensioni non rappresentano una minaccia per la stabilità del sistema. Questo è in realtà il modo in cui Amazon Route 53 gestisce i controlli dell'integrità degli endpoint (come gli indirizzi IP) per stabilire come gli utenti finali vengono instradati verso di loro.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

- Esegui un lavoro costante in modo che i sistemi non presentino guasti quando si verificano cambiamenti rapidi e significativi nel carico.
- Implementazione di dipendenze con accoppiamento debole Le dipendenze come sistemi di accodamento, sistemi di streaming, flussi di lavoro e bilanciatori del carico sono con accoppiamento debole. L'accoppiamento debole aiuta a isolare il comportamento di un componente dagli altri componenti che dipendono da esso, aumentando la resilienza e l'agilità.
  - The Amazon Builders' Library: Reliability, constant work, and a good cup of coffee

- AWS re:Invent 2018: Chiudere i circuiti e aprire le menti: come prendere il controllo dei sistemi, grandi e piccoli (include un lavoro costante) ARC337
  - Per l'esempio di un sistema di controllo dell'integrità che monitora 100.000 server, progetta i carichi di lavoro in modo che le dimensioni dei payload rimangano costanti indipendentemente dal numero di successi o di fallimenti.

Risorse

Documenti correlati:

- AmazonEC2: garantire l'idempotenza
- The Amazon Builders' Library: difficoltà dei sistemi distribuiti
- The Amazon Builders' Library: Reliability, constant work, and a good cup of coffee

Video correlati:

- AWS New York Summit 2019: introduzione alle architetture basate sugli eventi e ad Amazon (05) EventBridge MAD2
- AWS re:Invent 2018: Chiudere i circuiti e aprire le menti: come assumere il controllo di sistemi, grandi e piccoli (include un lavoro costante) ARC337
- AWS re:Invent 2018: Close Loops and Opening Minds: Come prendere il controllo dei sistemi, grandi e piccoli ARC337 (include accoppiamento libero, lavoro costante, stabilità statica)
- AWS re:Invent 2019: Passaggio ad architetture basate sugli eventi (08) SVS3

REL04-BP04 Rendere idempotenti le operazioni di mutazione

Un servizio idempotente promette che ogni richiesta venga elaborata esattamente una volta, in modo che effettuare più richieste identiche abbia lo stesso effetto di una singola richiesta. Questo rende più facile per un client implementare i tentativi senza temere che una richiesta venga erroneamente elaborata più volte. A tal fine, i client possono inviare richieste API con un token di idempotenza, che viene utilizzato ogni volta che la richiesta viene ripetuta. Un'API del servizio idempotente utilizza il token per restituire una risposta identica a quella restituita la prima volta che la richiesta è stata completata, anche se lo stato sottostante del sistema è cambiato.

In un sistema distribuito, è relativamente semplice eseguire un'azione al massimo una volta (il client effettua una sola richiesta) o almeno una volta (continuando a richiedere fino a quando il client non

riceve una conferma di successo). È più difficile garantire che un'azione venga eseguita esattamente una volta, in modo che più richieste identiche abbiano lo stesso effetto di una singola richiesta. Utilizzando i token di idempotenza nelle API, i servizi possono ricevere una richiesta di mutazione una o più volte senza la necessità di creare record duplicati o effetti collaterali.

Risultato desiderato: hai un approccio coerente, ben documentato e ampiamente adottato per garantire l'idempotenza in tutti i componenti e servizi.

Anti-pattern comuni:

- Applichi l'idempotenza indiscriminatamente, anche quando non è necessaria.
- Introduci una logica troppo complessa per implementare l'idempotenza.
- Utilizzi i timestamp come chiavi per l'idempotenza. Questo può causare imprecisioni a causa del disallineamento dell'orologio o a causa di più client che utilizzano gli stessi timestamp per applicare le modifiche.
- Archivi interi payload per l'idempotenza. In questo approccio, salvi i payload completi dei dati per ogni richiesta e li sovrascrivi a ogni nuova richiesta. Questo può ridurre le prestazioni e influenzare la scalabilità.
- Generi le chiavi in modo incoerente tra i vari servizi. Senza chiavi coerenti, i servizi potrebbero non riconoscere le richieste duplicate, con conseguenti risultati indesiderati.

Vantaggi dell'adozione di questa best practice:

- Maggiore scalabilità: il sistema può gestire tentativi e richieste duplicate senza dover eseguire una logica aggiuntiva o una complessa gestione dello stato.
- Maggiore affidabilità: l'idempotenza aiuta i servizi a gestire più richieste identiche in modo coerente, riducendo il rischio di effetti collaterali indesiderati o di record duplicati. Questo aspetto è particolarmente importante nei sistemi distribuiti, dove gli errori di rete e i tentativi sono frequenti.
- Miglioramento della coerenza dei dati: poiché la stessa richiesta produce la stessa risposta, l'idempotenza aiuta a mantenere la coerenza dei dati nei sistemi distribuiti. Questo è essenziale per mantenere l'integrità delle transazioni e delle operazioni.
- Gestione degli errori: i token di idempotenza rendono più semplice la gestione degli errori. Se un client non riceve una risposta a causa di un problema, può tranquillamente inviare nuovamente la richiesta con lo stesso token di idempotenza.

- Trasparenza operativa: l'idempotenza consente di migliorare il monitoraggio e la registrazione. I servizi possono registrare le richieste con i propri token di idempotenza, il che facilita il monitoraggio e il debug dei problemi.
- Contratto API semplificato: può semplificare il contratto tra i sistemi lato client e lato server e ridurre il timore di un'elaborazione errata dei dati.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

In un sistema distribuito, eseguire un'azione al massimo una volta (il client effettua una sola richiesta) o almeno una volta (il client continua a richiedere finché non viene confermato l'esito positivo) è relativamente semplice. Tuttavia, è difficile implementare esattamente una volta il comportamento. A tal fine, i client devono generare e fornire un token di idempotenza per ogni richiesta.

Utilizzando i token di idempotenza, un servizio può distinguere tra nuove richieste e richieste ripetute. Quando un servizio riceve una richiesta con un token di idempotenza, controlla se il token è già stato utilizzato. Se il token è stato utilizzato, il servizio recupera e restituisce la risposta memorizzata. Se il token è nuovo, il servizio elabora la richiesta, memorizza la risposta insieme al token e restituisce la risposta. Questo meccanismo rende tutte le risposte idempotenti, migliorando l'affidabilità e la coerenza del sistema distribuito.

Anche l'idempotenza è un comportamento importante delle architetture basate su eventi. Queste architetture sono in genere supportate da una coda di messaggi come Amazon SQS, Amazon MQ, Amazon Kinesis Streams o Amazon Managed Streaming per Apache Kafka (MSK). In alcune circostanze, un messaggio pubblicato una sola volta può essere accidentalmente recapitato più di una volta. Quando un publisher genera e include i token di idempotenza nei messaggi, richiede che l'elaborazione di qualsiasi messaggio duplicato ricevuto non comporti un'azione ripetuta per lo stesso messaggio. I consumatori devono tenere traccia di ogni token ricevuto e ignorare i messaggi che contengono token duplicati.

I servizi e i consumatori devono anche passare il token di idempotenza ricevuto a tutti i servizi a valle che vengono chiamati. Ogni servizio a valle della catena di elaborazione ha la stessa responsabilità di assicurarsi che l'idempotenza sia implementata per evitare l'effetto collaterale di elaborare un messaggio più di una volta.

Passaggi dell'implementazione

1. Identifica le operazioni idempotenti

Determina quali operazioni richiedono idempotenza. Questi includono in genere i metodi HTTP POST, PUT e DELETE e le operazioni di inserimento, aggiornamento o eliminazione del database. Le operazioni che non mutano lo stato, come le query di sola lettura, di solito non richiedono idempotenza, a meno che non abbiano effetti collaterali.

2. Utilizza identificatori univoci

   Includi un token univoco in ogni richiesta di operazione idempotente inviata dal mittente, direttamente nella richiesta o come parte dei relativi metadati (ad esempio, un'intestazione HTTP). Ciò consente al destinatario di riconoscere e gestire richieste o operazioni duplicate. Gli identificatori comunemente usati per i token includono Universally Unique Identifier (UUID) e K-Sortable Unique Identifier (KSUID).

3. Monitora e gestisci lo stato

   Mantieni lo stato di ogni operazione o richiesta nel carico di lavoro. Ciò può essere ottenuto memorizzando il token di idempotenza e lo stato corrispondente (come in attesa, completato o non riuscito) in un database, una cache o un altro archivio persistente. Queste informazioni sullo stato consentono al carico di lavoro di identificare e gestire richieste o operazioni duplicate.

   Mantieni la coerenza e l'atomicità utilizzando, se necessario, meccanismi di controllo della simultaneità appropriati, come blocchi, transazioni o controlli ottimistici della simultaneità. Questo include il processo di registrazione del token idempotente e l'esecuzione di tutte le operazioni di mutazione associate al servizio della richiesta. Questo aiuta a prevenire le condizioni di gara e verifica che le operazioni idempotenti vengano eseguite correttamente.

   Rimuovi periodicamente i vecchi token di idempotenza dal datastore per gestire l'archiviazione e le prestazioni. Se il sistema di archiviazione lo supporta, prendi in considerazione l'utilizzo di timestamp di scadenza per i dati (spesso noti come valori di time-to-live, o TTL). La probabilità di riutilizzo dei token di idempotenza diminuisce nel tempo.

   Le comuni opzioni di archiviazione AWS genericamente utilizzate per memorizzare i token di idempotenza e il relativo stato includono:

   - Amazon DynamoDB: DynamoDB è un servizio di database NoSQL che offre prestazioni a bassa latenza ed elevata disponibilità, il che lo rende adatto all'archiviazione di dati correlati all'idempotenza. Il modello di dati chiave-valore e documento di DynamoDB consente di memorizzare e recuperare in modo efficiente i token di idempotenza e le informazioni di stato associate. DynamoDB può anche far scadere automaticamente i token di idempotenza se l'applicazione imposta un valore TTL al momento dell'inserimento.

- Amazon ElastiCache: ElastiCache è in grado di archiviare token di idempotenza con un elevato throughput, bassa latenza e a costo ridotto. ElastiCache (Redis) e ElastiCache (Memcached) possono entrambi far scadere automaticamente i token di idempotenza se l'applicazione imposta un valore TTL al momento dell'inserimento.
- Amazon Relational Database Service (RDS): puoi utilizzare Amazon RDS per archiviare i token di idempotenza e le informazioni di stato correlate, soprattutto se l'applicazione utilizza già un database relazionale per altri scopi.
- Amazon Simple Storage Service (S3): Amazon S3 è un servizio di archiviazione di oggetti altamente scalabile e durevole che può essere utilizzato per archiviare i token di idempotenza e i metadati correlati. Le funzionalità di controllo delle versioni di S3 possono essere particolarmente utili per il mantenimento dello stato di operazioni idempotenti. La scelta del servizio di archiviazione dipende in genere da fattori quali il volume dei dati correlati all'idempotenza, le caratteristiche di prestazione richieste, l'esigenza di durabilità e disponibilità e il modo in cui il meccanismo di idempotenza si integra con l'architettura complessiva del carico di lavoro.

4. Implementa operazioni idempotenti

   Progetta l'API e i componenti del carico di lavoro in modo che siano idempotenti. Incorpora i controlli di idempotenza nei componenti del carico di lavoro. Prima di elaborare una richiesta o eseguire un'operazione, controlla se l'identificatore univoco è già stato elaborato. In caso affermativo, restituisci il risultato precedente invece di eseguire nuovamente l'operazione. Ad esempio, se un client invia una richiesta per creare un utente, verifica se esiste già un utente con lo stesso identificatore univoco. Se l'utente esiste, deve restituire le informazioni utente esistenti anziché crearne una nuovo. Allo stesso modo, se un consumatore di code riceve un messaggio con un token di idempotenza duplicato, deve ignorare il messaggio.

   Crea suite di test complete che convalidino l'idempotenza delle richieste. Devono coprire un'ampia gamma di scenari, come richieste andate a buon fine, richieste non riuscite e richieste duplicate.

   Se il carico di lavoro sfrutta le funzioni AWS Lambda, prendi in considerazione Powertools per AWS Lambda. Powertools per AWS Lambda è un kit di strumenti per sviluppatori che aiuta a implementare le best practice serverless e ad aumentare la velocità di sviluppo quando si lavora con le funzioni AWS Lambda Lambda. In particolare, fornisce un'utilità per convertire le funzioni Lambda in operazioni idempotenti, sicure per riprovare.

5. Comunica chiaramente l'idempotenza

Documenta l'API e i componenti del carico di lavoro per comunicare chiaramente la natura idempotente delle operazioni. Questo aiuta i clienti a comprendere il comportamento previsto e a capire come interagire con il carico di lavoro in modo affidabile.

6. Monitora ed esegui audit

   Implementa meccanismi di monitoraggio e audit per rilevare eventuali problemi correlati all'idempotenza delle risposte, come variazioni impreviste delle risposte o gestione eccessiva di richieste duplicate. Questo può aiutare a rilevare e indagare su eventuali problemi o comportamenti imprevisti nel carico di lavoro.

Risorse

Best practice correlate:

- REL05-BP03 Controllo e limitazione delle chiamate di ripetizione
- REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)
- REL06-BP03 Invio di notifiche (elaborazione e avvisi in tempo reale)
- REL08-BP02 Esecuzione di test funzionali come parte integrante dell'implementazione

Documenti correlati:

- The Amazon Builders' Library: Making retries safe with idempotent APIs
- The Amazon Builders' Library: difficoltà dei sistemi distribuiti
- The Amazon Builders' Library: Reliability, constant work, and a good cup of coffee
- Amazon Elastic Container Service: Ensuring idempotency
- Come faccio a rendere idempotente la mia funzione Lambda?
- Garantire l'idempotenza nelle richieste Amazon EC2 API

Video correlati:

- Building Distributed Applications with Event-driven Architecture - AWS Online Tech Talks
- AWS re:Invent 2.023 - Building next-generation applications with event-driven architecture
- AWS re:Invent 2.023 - Advanced integration patterns & trade-offs for loosely coupled systems
- AWS re:Invent 2.023 - Advanced event-driven patterns with Amazon EventBridge

- AWS re:Invent 2.018 - Close Loops and Opening Minds: How to Take Control of Systems, Big and Small ARC337 (includes loose coupling, constant work, static stability)
- AWS re:Invent 2.019 - Moving to event-driven architectures (SVS308)

Strumenti correlati:

- Idempotenza con AWS Lambda Powertools (Java)
- Idempotenza con AWS Lambda Powertools (Python)
- Pagina GitHub AWS Lambda Powertools

REL 5. Come si progettano le interazioni in un sistema distribuito per mitigare o affrontare gli errori?

I sistemi distribuiti si basano sulle reti di comunicazione per interconnettere i componenti (ad esempio server o servizi). Il carico di lavoro deve funzionare in modo affidabile nonostante la perdita o la latenza dei dati su queste reti. I componenti del sistema distribuito devono funzionare in modo da non influire negativamente su altri componenti o sul carico di lavoro. Queste best practice permettono ai carichi di lavoro di tollerare le sollecitazioni o i guasti, recuperare più rapidamente e mitigare l'impatto di tali problemi. Il risultato è un miglioramento del tempo medio di ripristino (MTTR).

Best practice

- REL05-BP01 Implementazione della normale riduzione delle prestazioni per trasformare le dipendenze forti applicabili in dipendenze deboli
- REL05-BP02 Richieste di limitazione (della larghezza di banda della rete)
- REL05-BP03 Controlla e limita le chiamate di nuovo tentativo
- REL05-BP04 Anticipare l’errore (fail fast) e limitare le code
- REL05-BP05 Imposta i timeout dei client
- REL05-BP06 Rendere i sistemi senza stato ove possibile
- REL05-BP07 Implementare leve di emergenza

REL05-BP01 Implementazione della normale riduzione delle prestazioni per trasformare le dipendenze forti applicabili in dipendenze deboli

I componenti dell'applicazione devono continuare a svolgere la loro funzione principale anche se le dipendenze non sono disponibili. Potrebbero fornire dati leggermente obsoleti, dati alternativi o

addirittura nessun dato. Ciò garantisce che la funzionalità complessiva del sistema sia ostacolata solo in minima parte da errori localizzati, garantendo al contempo il valore aziendale intrinseco.

Risultato desiderato: quando le dipendenze di un componente non sono integre, il componente stesso può comunque funzionare, anche se in modo degradato. Le modalità di errore dei componenti devono essere considerate come funzionamenti normali. I flussi di lavoro devono essere progettati in modo tale che questi errori non conducano a un fallimento completo o almeno a stati prevedibili e recuperabili.

Anti-pattern comuni:

- Mancata identificazione della funzionalità aziendale di base necessaria. Mancata verifica del funzionamento dei componenti anche in caso di errori di dipendenza.
- Mancata restituzione di dati sugli errori o quando solo una delle dipendenze non è disponibile e possono comunque essere restituiti risultati parziali.
- Creazione di uno stato incoerente quando una transazione non va a buon fine parzialmente.
- Mancata disponibilità di alternative per accedere a un archivio di parametri centralizzato.
- Invalidare o svuotare lo stato locale a seguito di un aggiornamento non riuscito senza considerare le conseguenze di tale operazione.

Vantaggi dell'adozione di questa best practice: la normale riduzione delle prestazioni migliora la disponibilità del sistema nel suo complesso e conserva la funzionalità delle funzioni più importanti anche in caso di errori.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

L'implementazione di una normale riduzione delle prestazioni aiuta a ridurre al minimo l'impatto degli errori di dipendenza sul funzionamento dei componenti. Idealmente, un componente rileva gli errori nelle dipendenze e trova soluzioni alternative in modo da avere un impatto minimo sugli altri componenti o clienti.

Progettare per una normale riduzione delle prestazioni significa considerare le potenziali modalità di errore durante la progettazione delle dipendenze. Per ogni modalità di errore, disponi di un modo per fornire la maggior parte delle funzionalità o almeno quelle più critiche del componente a chiamanti o clienti. Queste considerazioni possono diventare requisiti aggiuntivi testabili e verificabili. Idealmente,

un componente è in grado di svolgere la sua funzione principale in modo accettabile anche in caso di errore di una o più dipendenze.

Questa è una discussione di carattere tanto commerciale quanto tecnico. Tutti i requisiti aziendali sono importanti e devono essere soddisfatti, se possibile. Tuttavia, ha ancora senso chiedersi cosa dovrebbe succedere quando non tutti i requisiti possono essere soddisfatti. Un sistema può essere progettato per essere disponibile e coerente, ma nelle circostanze in cui è necessario eliminare un requisito, qual è quello più importante? Per l'elaborazione dei pagamenti, potrebbe essere la coerenza. Per un'applicazione in tempo reale, potrebbe essere la disponibilità. Per un sito Web rivolto ai clienti, la risposta può dipendere dalle aspettative dei clienti.

Il significato di ciò dipende dai requisiti del componente e da ciò che dovrebbe essere considerato la sua funzione principale. Esempio:

- Un sito di e-commerce potrebbe visualizzare dati provenienti da più sistemi diversi, come consigli personalizzati, prodotti con il punteggio più alto e lo stato degli ordini dei clienti sulla pagina di destinazione. Quando in un sistema upstream si verifica un errore, ha comunque senso mostrare tutto il resto, invece di mostrare una pagina di errore a un cliente.
- Un componente che esegue la scrittura in batch può continuare a elaborare un batch se una delle singole operazioni fallisce. Dovrebbe essere semplice implementare un meccanismo di ripetizione dei tentativi. A tale scopo, è sufficiente restituire al chiamante informazioni su quali operazioni hanno avuto successo, quali e perché non sono riuscite, oppure inserendo le richieste non riuscite in una coda DLQ per implementare nuovi tentativi asincroni. Anche le informazioni sulle operazioni non riuscite devono essere registrate.
- Un sistema che elabora le transazioni deve verificare che vengano eseguiti tutti gli aggiornamenti o nessun aggiornamento. Per le transazioni distribuite, il modello Saga può essere utilizzato per ripristinare le operazioni precedenti nel caso in cui fallisca un'operazione successiva della stessa transazione. Qui, la funzione principale è mantenere la coerenza.
- I sistemi critici dal punto di vista temporale dovrebbero essere in grado di gestire le dipendenze che non rispondono in modo tempestivo. In questi casi, è possibile utilizzare lo schema dell'interruttore. Quando inizia a verificarsi il timeout delle risposte di una dipendenza, il sistema può passare a uno stato chiuso in cui non vengono effettuate chiamate aggiuntive.
- Un'applicazione può leggere i parametri da un archivio di parametri. Può essere utile creare immagini di container con un set di parametri predefinito e utilizzarli nel caso in cui l'archivio dei parametri non sia disponibile.

Si noti che i percorsi seguiti in caso di errore dei componenti devono essere testati e devono essere significativamente più semplici del percorso primario. In genere, è consigliabile evitare strategie di fallback.

Passaggi dell'implementazione

Identifica le dipendenze esterne e interne. Considera i tipi di errore che si possono verificare nelle dipendenze. Considera i modi per ridurre al minimo l'impatto negativo sui sistemi upstream e downstream e sui clienti durante questi errori.

Di seguito è riportato un elenco di dipendenze e di come ridurre normalmente le prestazioni quando si verifica un errore a livello di dipendenze:

1. Errore parziale delle dipendenze: un componente può effettuare più richieste ai sistemi downstream, sia come richieste multiple a un sistema sia come richiesta a più sistemi. A seconda del contesto aziendale, possono essere appropriate diverse modalità di gestione (per maggiori dettagli, consulta gli esempi precedenti nella Guida all'implementazione).
2. Un sistema downstream non è in grado di elaborare le richieste a causa del carico elevato: se le richieste rivolte a un sistema downstream non vanno costantemente a buon fine, non ha senso continuare a riprovare. Ciò può creare un carico aggiuntivo su un sistema già sovraccarico e rendere più difficile il ripristino. Qui è possibile utilizzare lo schema dell'interruttore, che monitora le chiamate non riuscite a un sistema downstream. Se un numero elevato di chiamate ha esito negativo, interromperà l'invio di altre richieste al sistema downstream e solo occasionalmente lascerà passare le chiamate per verificare se il sistema downstream è nuovamente disponibile.
3. Un archivio di parametri non è disponibile: per trasformare un archivio di parametri, è possibile utilizzare la cache delle dipendenze a protezione debole o i valori predefiniti integri inclusi nelle immagini del container o del computer. Tieni presente che queste impostazioni predefinite devono essere costantemente aggiornate e incluse nelle suite di test.
4. Un servizio di monitoraggio o altra dipendenza non funzionale non è disponibile: se un componente non è in grado di inviare a intermittenza log, metriche o tracce a un servizio di monitoraggio centralizzato, spesso è meglio continuare a eseguire le funzioni aziendali come al solito. Non registrare o eseguire il push delle metriche in modo invisibile all'utente per un lungo periodo di tempo spesso non è una procedura accettabile. Inoltre, in alcuni casi d'uso potrebbero essere necessari dati di controllo completi per soddisfare i requisiti di conformità.
5. Un'istanza primaria di un database relazionale potrebbe non essere disponibile: come quasi tutti i database relazionali, Amazon Relational Database Service può presentare solo un'istanza di scrittura primaria. Questo crea un unico punto di errore per i carichi di lavoro di scrittura e rende

più difficile il dimensionamento. Questo problema può essere parzialmente mitigato utilizzando una configurazione Multi-AZ per una disponibilità elevata o Amazon Aurora Serverless per un migliore dimensionamento. Per requisiti di disponibilità molto elevati, può essere logico non fare affatto affidamento sull'istanza di scrittura primaria. Per le query che si limitano a leggere, è possibile utilizzare repliche di lettura, che forniscono ridondanza e la possibilità di aumentare orizzontalmente e anche verticalmente. Le operazioni di scrittura possono essere memorizzate nel buffer, ad esempio in una coda Amazon Simple Queue Service, in modo che le richieste di scrittura dei clienti possano comunque essere accettate anche se l'istanza primaria non è temporaneamente disponibile.

Risorse

Documenti correlati:

- Amazon API Gateway: Throttle API Requests for Better Throughput
- CircuitBreaker (riepilogo dell'interruttore dal libro "Release It!")
- Ripetizione dei tentativi in caso di errore e backoff esponenziale in AWS
- Michael Nygard "Release It! Design and Deploy Production-Ready Software"
- The Amazon Builders' Library: evitare il fallback nei sistemi distribuiti
- The Amazon Builders' Library: evitare insormontabili backlog di code
- The Amazon Builders' Library: sfide e strategie del caching
- The Amazon Builders' Library: timeout, nuovi tentativi e backoff con jitter

Video correlati:

- Retry, backoff, and jitter: AWS re:Invent 2019: Introducing The Amazon Builders' Library (DOP328)

REL05-BP02 Richieste di limitazione (della larghezza di banda della rete)

Usa le richieste di limitazione (della larghezza di banda della rete) per mitigare l'esaurimento delle risorse dovuto ad aumenti imprevisti della domanda. Le richieste inferiori alla percentuale di limitazione (della larghezza di banda della rete) vengono elaborate, mentre quelle che superano il limite definito vengono rifiutate con un messaggio che indica che la richiesta è stata limitata.

Risultato desiderato: i picchi di volume di grandi dimensioni dovuti a improvvisi aumenti del traffico dei clienti, attacchi di flooding o tempeste di ripetizioni dei tentativi sono mitigati dalla limitazione

(della larghezza di banda della rete) delle richieste, che consente ai carichi di lavoro di continuare la normale elaborazione del volume di richieste supportato.

Anti-pattern comuni:

- Le limitazioni (della larghezza di banda della rete) degli endpoint API non sono implementate o vengono implementate in base ai valori predefiniti senza considerare i volumi previsti.
- Gli endpoint delle API non sono sottoposti a test di carico né i limiti relativi alla limitazione (della larghezza di banda della rete) vengono testati.
- Limitazione (della larghezza di banda della rete) dei tassi di richiesta senza considerare le dimensioni o la complessità delle richieste.
- Verifica delle percentuali massime di richieste o delle dimensioni massime delle richieste, senza però testarle congiuntamente.
- Le risorse non vengono allocate entro gli stessi limiti stabiliti durante i test.
- I piani di utilizzo non sono stati configurati o considerati per gli utenti di API Application to Application (A2A).
- Gli utenti di code con scalabilità orizzontale non hanno configurato le impostazioni di simultaneità massima.
- La limitazione della velocità per indirizzo IP non è stata implementata.

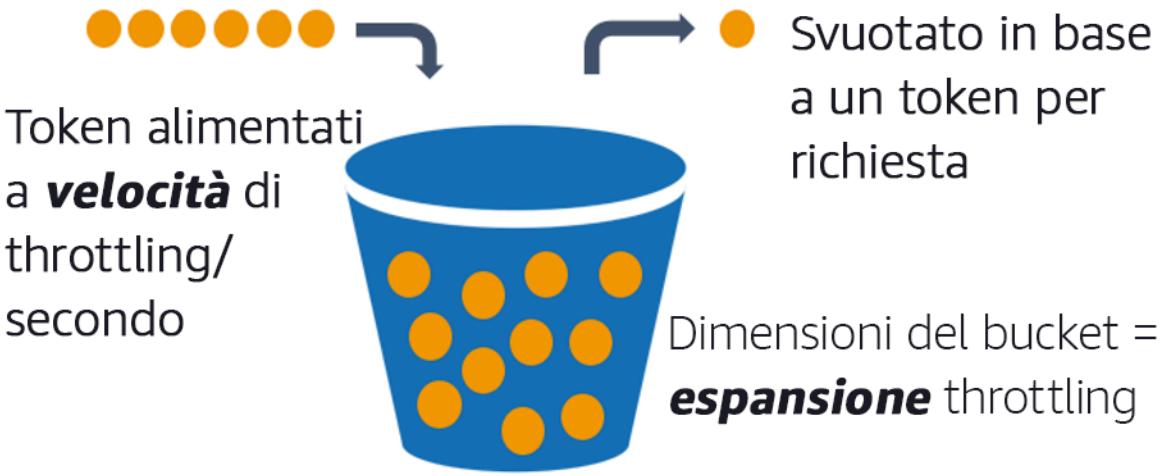
Vantaggi dell'adozione di questa best practice: i carichi di lavoro che stabiliscono limiti di accelerazione sono in grado di funzionare normalmente ed elaborare correttamente il caricamento delle richieste accettate in presenza di picchi di volume imprevisti. I picchi improvvisi o prolungati di richieste alle API e alle code vengono applicate limitazioni (della larghezza di banda della rete) e non esauriscono le risorse di elaborazione delle richieste. La limitazione (della larghezza di banda della rete) limita i singoli richiedenti in modo che gli alti volumi di traffico provenienti da un singolo indirizzo IP o da un consumatore di API non esauriscano le risorse che impattano sugli altri consumatori.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I servizi devono essere progettati per elaborare una capacità nota di richieste; tale capacità può essere stabilita mediante test di carico. Se le percentuali di arrivo delle richieste superano i limiti, la risposta appropriata segnala che una richiesta ha subito la limitazione (della larghezza di banda della rete). Ciò consente all'utente di gestire l'errore e riprovare in un secondo momento.

Quando il servizio richiede un'implementazione della limitazione (della larghezza di banda della rete), prendi in considerazione l'implementazione dell'algoritmo token bucket, in cui un token conta come una richiesta. I token vengono alimentati a una specifica velocità di limitazione (della larghezza di banda della rete) al secondo e svuotati in modo asincrono in base a un token per richiesta.



Algoritmo token bucket.

Gateway Amazon API implementa l'algoritmo token bucket in base ai limiti dell'account e della regione e può essere configurato per cliente con piani di utilizzo. Inoltre, Amazon Simple Queue Service (Amazon SQS) e Amazon Kinesis possono memorizzare in buffer le richieste in modo da ridurre la frequenza delle richieste e consentire percentuali di limitazione (della larghezza di banda della rete) più elevate per le richieste che è possibile soddisfare. Infine, puoi implementare limitazioni della velocità con AWS WAF per la limitazione (della larghezza di banda della rete) di specifici utenti di API che generano un carico insolitamente elevato.

Passaggi dell'implementazione

Puoi configurare API Gateway con limiti relativi alla limitazione (della larghezza di banda della rete) per le tue API e restituire errori `429 Too Many Requests` in caso di superamento dei limiti. Puoi utilizzare AWS WAF con gli endpoint AWS AppSync e API Gateway per abilitare la limitazione della velocità per indirizzo IP. Inoltre, laddove il sistema può tollerare l'elaborazione asincrona, è possibile inserire i messaggi in una coda o in un flusso per velocizzare le risposte ai client del servizio, il che consente di aumentare i tassi di limitazione (della larghezza di banda della rete).

Grazie all'elaborazione asincrona, dopo aver configurato Amazon SQS come origine di eventi per AWS Lambda, puoi configurare la simultaneità massima per evitare che percentuali elevate di eventi

consumino la quota di esecuzione simultanea disponibile dell'account necessaria ad altri servizi nel carico di lavoro o nell'account.

Sebbene API Gateway fornisca un'implementazione gestita dell'algoritmo token bucket, nei casi in cui non sia possibile utilizzare API Gateway, puoi sfruttare le implementazioni open source specifiche del linguaggio (consulta gli esempi correlati nella sezione Risorse) dell'algoritmo token bucket per i tuoi servizi.

- Analizza e configura i limiti relativi alla limitazione (della larghezza di banda della rete) di API Gateway a livello di account per regione, API per fase e chiave API per livelli del piano di utilizzo.
- Applica regole di limitazione della velocità AWS WAF all'API Gateway e agli endpoint AWS AppSync per proteggerti dagli attacchi flood e bloccare gli IP dannosi. Le regole di limitazione (della larghezza di banda della rete) possono anche essere configurate su chiavi API AWS AppSync per gli utenti A2A.
- Valuta se ti occorre un controllo maggiore sulla limitazione (della larghezza di banda della rete) rispetto al controllo sulla limitazione della velocità per le API AWS AppSync. In tal caso, configura un'API Gateway prima dell'endpoint AWS AppSync.
- Se le code Amazon SQS sono configurate come trigger per gli utenti delle code Lambda, imposta la simultaneità massima su un valore che garantisca un'elaborazione sufficiente da soddisfare i tuoi obiettivi di livello di servizio, senza consumare limiti di simultaneità che influiscono su altre funzioni Lambda. Valuta la possibilità di impostare la simultaneità riservata su altre funzioni Lambda nello stesso account e nella stessa regione quando utilizzi le code con Lambda.
- Utilizza API Gateway con integrazioni di servizi native per Amazon SQS o Kinesis per memorizzare le richieste nel buffer.
- Se non puoi utilizzare API Gateway, consulta le librerie specifiche della lingua per implementare l'algoritmo token bucket per il tuo carico di lavoro. Controlla la sezione degli esempi e cerca una libreria adatta.
- Verifica i limiti che intendi impostare o che prevedi di incrementare e documenta i limiti testati.
- Non aumentare i limiti oltre i valori stabiliti durante i test. Quando si aumenta un limite, verifica che le risorse allocate siano equivalenti o superiori a quelle degli scenari di test prima di applicare l'aumento.

Risorse

Best practice correlate:

- REL04-BP03 Fai un lavoro costante
- REL05-BP03 Controlla e limita le chiamate di nuovo tentativo

Documenti correlati:

- Amazon API Gateway: Throttle API Requests for Better Throughput
- AWS WAF: Rate-based rule statement
- Introducing maximum concurrency of AWS Lambda when using Amazon SQS as an event source
- AWS Lambda: Maximum Concurrency

Esempi correlati:

- The three most important AWS WAF rate-based rules
- Java Bucket4j
- Algoritmo token bucket per Python
- Algoritmo token bucket a livello di nodo
- Limitazione della velocità di threading del sistema .NET

Video correlati:

- Implementing GraphQL API security best practices with AWS AppSync

Strumenti correlati:

- Gateway Amazon API
- AWS AppSync
- Amazon SQS
- Amazon Kinesis
- AWS WAF
- Sala d'attesa virtuale su AWS

REL05-BP03 Controlla e limita le chiamate di nuovo tentativo

Utilizza il backoff esponenziale per rieseguire le richieste a intervalli progressivamente più lunghi tra i singoli nuovi tentativi. Introduci il jitter tra i tentativi per la randomizzazione degli intervalli di ripetizione. Limita il numero massimo di tentativi.

Risultato desiderato: i componenti tipici di un sistema software distribuito includono server, sistemi di bilanciamento del carico, database e server. DNS Durante il normale funzionamento, questi componenti possono rispondere alle richieste con errori temporanei o limitati e anche errori che sarebbero persistenti indipendentemente dai nuovi tentativi. Quando i client effettuano richieste ai servizi, le richieste consumano risorse tra cui memoria, thread, connessioni, porte o altre risorse limitate. Controllare e limitare i nuovi tentativi è una strategia per liberare risorse e ridurne al minimo il consumo in modo che i componenti del sistema sottoposti a stress non vengano sovraccaricati.

Quando vanno in timeout o ricevono risposte di errore, le richieste client devono decidere se eseguire o meno nuovi tentativi. Se vengono eseguiti nuovi tentativi, questi verranno eseguiti con un backoff esponenziale con jitter e un numero massimo di tentativi. Di conseguenza, i servizi e i processi backend riducono il carico e i tempi di riparazione automatica, con un conseguente ripristino più rapido e una corretta gestione delle richieste.

Anti-pattern comuni:

- Implementazione di nuovi tentativi senza aggiungere il backoff esponenziale, il jitter e il numero massimo di tentativi. Il backoff e il jitter aiutano a evitare picchi di traffico artificiali dovuti a tentativi involontariamente coordinati a intervalli standard.
- Implementazione di nuovi tentativi senza testarne gli effetti o presupponendo che i nuovi tentativi siano già integrati in scenari di ripetizione e senza test. SDK
- La mancata comprensione dei codici di errore pubblicati nelle dipendenze porta a ritentare tutti gli errori, compresi quelli la cui causa è chiara e indica la mancanza di autorizzazione, un errore di configurazione o un'altra condizione che prevedibilmente non si risolverà senza un intervento manuale.
- Mancata gestione delle best practice di osservabilità, compresi il monitoraggio e la segnalazione di ripetuti guasti del servizio, in modo che i problemi sottostanti siano resi noti e possano essere risolti.
- Sviluppo di meccanismi di ripetizione personalizzati quando le funzionalità di ripetizione dei tentativi integrate o di terze parti sono sufficienti.
- Riprovare su più livelli dello stack di applicazioni in modo da accrescere in modo significativo i nuovi tentativi e pertanto da consumare ulteriormente le risorse in una tempesta di ripetizioni dei

tentativi. Assicurati di comprendere in che modo questi errori influiscono sulla tua applicazione e sulle dipendenze su cui fai affidamento, quindi implementa i nuovi tentativi a un solo livello.

- Riesecuzione delle chiamate dei servizi non idempotenti, con effetti collaterali imprevisti come risultati duplicati.

Vantaggi dell'adozione di questa best practice: i nuovi tentativi aiutano i client a ottenere i risultati desiderati quando le richieste non vanno a buon fine, ma consumano più tempo del server per ottenere le risposte corrette desiderate. Quando gli errori sono rari o transitori, i nuovi tentativi funzionano correttamente. Quando gli errori sono causati da un sovraccarico di risorse, i nuovi tentativi possono peggiorare le cose. L'aggiunta di un backoff esponenziale con jitter ai tentativi dei client consente ai server di recuperare risorse quando gli errori sono causati dal sovraccarico delle risorse. Il jitter evita l'allineamento delle richieste in picchi e il backoff riduce l'aumento del carico causato dall'aggiunta di nuovi tentativi al normale carico delle richieste. Infine, è importante configurare un numero massimo di tentativi o il tempo trascorso per evitare la creazione di backlog che producono errori metastabili.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Controlla e limita le chiamate riproposte. Utilizza il backoff esponenziale per eseguire nuovi tentativi dopo intervalli progressivamente più lunghi. Introduci il jitter per la randomizzazione degli intervalli di ripetizione e limitare il numero massimo di tentativi.

Per impostazione predefinita, alcuni AWS SDKs implementano nuovi tentativi e il backoff esponenziale. Utilizza queste AWS implementazioni integrate laddove applicabile nel tuo carico di lavoro. Implementa una logica simile nel tuo carico di lavoro nelle chiamate di servizi idempotenti e i cui tentativi migliorano la disponibilità dei client. Potrai decidere quali sono i timeout e quando cessare i tentativi in base al tuo caso d'uso. Crea ed esegui scenari di test per quei casi d'uso relativi ai nuovi tentativi.

Passaggi dell'implementazione

- Determina il livello ottimale nello stack di applicazioni per implementare nuovi tentativi per i servizi su cui si basa l'applicazione.
- Sii consapevole delle strategie esistenti SDKs che implementano strategie di ripetizione comprovate con backoff e jitter esponenziali per il linguaggio che preferisci, e preferiscile alla stesura di implementazioni personalizzate con nuovi tentativi.

- Verifica che i servizi siano caratterizzati dall'idempotenza prima dell'implementazione di nuovi tentativi. Una volta implementati i nuovi tentativi, assicurati che siano testati e che vengano regolarmente eseguiti in produzione.
- Quando chiami il AWS servizioAPIs, usa AWS SDKsand e AWS CLIcomprendi le opzioni di configurazione Retry. Determina se le impostazioni predefinite sono adatte al tuo caso d'uso, esegui i test e regola i valori secondo necessità.

Risorse

Best practice correlate:

- REL04-BP04 Rendere idempotenti le operazioni di mutazione
- REL05-BP02 Richieste di limitazione (della larghezza di banda della rete)
- REL05-BP04 Anticipare l’errore (fail fast) e limitare le code
- REL05-BP05 Imposta i timeout dei client
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

Documenti correlati:

- Errore, tentativi e backoff esponenziale in AWS
- The Amazon Builders' Library: timeout, nuovi tentativi e backoff con jitter
- Exponential Backoff and Jitter
- Rendere sicuri i nuovi tentativi con idempotent APIs

Esempi correlati:

- Spring Retry
- Resilience4j Retry

Video correlati:

- Riprova, backoff e jitter: AWS re:Invent 2019: Presentazione di The Amazon Builders' Library () DOP328

Strumenti correlati:

- AWS SDKsStrumenti e strumenti: Riprova il comportamento
- AWS Command Line Interface: AWS CLI riprova

REL05-BP04 Anticipare l'errore (fail fast) e limitare le code

Se un servizio non è in grado di rispondere correttamente a una richiesta, procede ad anticipare l'errore (fail fast). Ciò consente il rilascio delle risorse associate a una richiesta e permette al servizio di recuperare le risorse se queste sono in esaurimento. L'anticipazione degli errori (fail fast) è un modello di progettazione software consolidato che può essere usato per creare carichi di lavoro altamente affidabili nel cloud. Anche l'accodamento è un modello di integrazione aziendale consolidato che può semplificare il carico e consentire ai client di rilasciare risorse quando l'elaborazione asincrona può essere tollerata. Quando un servizio è in grado di rispondere correttamente in condizioni normali ma restituisce un esito negativo quando la frequenza delle richieste è troppo alta, utilizza una coda per memorizzare le richieste nel buffer. Tuttavia, non consentire la creazione di backlog di code lunghe che possono comportare l'elaborazione di richieste obsolete già dismesse dal client.

Risultato desiderato: quando i sistemi rilevano conflitti a livello di risorse, timeout, eccezioni o errori che rendono irraggiungibili gli obiettivi dei livelli di servizio, le strategie volte ad anticipare l'errore (fail fast) consentono un ripristino più rapido del sistema. I sistemi che devono assorbire i picchi di traffico e sono in grado di gestire l'elaborazione asincrona possono migliorare l'affidabilità consentendo ai client di rilasciare rapidamente le richieste utilizzando le code per archiviare le richieste nei servizi di backend. Quando le richieste vengono memorizzate nei buffer delle code, vengono implementate strategie di gestione delle code per evitare backlog ingestibili.

Anti-pattern comuni:

- Implementazione delle code di messaggi senza la configurazione delle code DLQ o degli allarmi nei volumi DLQ per rilevare quando un sistema è in errore.
- Mancata misurazione dell'età dei messaggi in una coda, misurazione della latenza per capire quando gli utenti della coda sono in ritardo o generano errori che causano un nuovo tentativo.
- Mancata cancellazione dei messaggi nel backlog da una coda quando non è più necessario elaborare questi messaggi se l'azienda non lo richiede più.
- La configurazione delle code First in First Out (FIFO) quando le code Last In First Out (LIFO) soddisferebbe meglio le esigenze dei client, ad esempio quando non sono richiesti ordini rigorosi e l'elaborazione dei backlog sta ritardando tutte le richieste nuove e urgenti, con conseguente violazione dei livelli di servizio per tutti i client.

- Esposizione delle code interne ai client, invece dell'esposizione delle API che gestiscono l'acquisizione del lavoro e l'inserimento delle richieste in code interne.
- Combinazione di un numero eccessivo di tipi di richieste di lavoro in un'unica coda: ciò può aggravare le condizioni dei backlog in seguito alla distribuzione delle richieste di risorse tra i tipi di richiesta.
- Elaborazione di richieste complesse e semplici nella stessa coda, nonostante siano necessari monitoraggio, timeout e allocazioni di risorse diversi.
- Mancata convalida degli input o utilizzo di asserzioni per implementare meccanismi in grado di anticipare l'errore (fail fast) nel software che generano eccezioni a componenti di livello superiore in grado di gestire normalmente gli errori.
- Mancata rimozione delle risorse in errore dall'instradamento delle richieste, soprattutto quando gli errori generano risultati sia positivi che negativi dovuti ad arresti anomali e riavvii, errori intermittenti a livello di dipendenze, capacità ridotta o perdita di pacchetti di rete.

Vantaggi dell'adozione di questa best practice: i sistemi in grado di anticipare l'errore (fail fast) sono più facili da sottoporre al debug e alla correzione degli errori e spesso presentano problemi di codifica e configurazione prima che le versioni vengano pubblicate in produzione. I sistemi che incorporano strategie di accodamento efficaci forniscono maggiore resilienza e affidabilità in caso di picchi di traffico e di condizioni intermittenti di errore del sistema.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Le strategie volte ad anticipare l'errore (fail fast) possono essere codificate in soluzioni software e configurate nell'infrastruttura. Oltre ad anticipare l'errore (fail fast), le code sono una tecnica semplice ma affidabile di definizione dell'architettura che consente il caricamento senza problemi di componenti disaccoppiati del sistema. Amazon CloudWatch fornisce funzionalità per il monitoraggio e la segnalazione di guasti. Una volta accertato il malfunzionamento di un sistema, è possibile richiamare strategie di mitigazione, ad esempio per evitare problemi dovuti a risorse danneggiate. Se i sistemi implementano code con Amazon SQS e altre tecnologie di accodamento, per semplificare il caricamento, devono valutare come gestire i backlog e gli errori di utilizzo dei messaggi.

Passaggi dell'implementazione

- Implementa asserzioni programmatiche o metriche specifiche nel tuo software e usale per ricevere avvisi espliciti in caso di problemi di sistema. Con Amazon CloudWatch puoi creare metriche e allarmi basati sul modello di log delle applicazioni e sulla strumentazione SDK.

- Usa le metriche CloudWatch e gli allarmi per eseguire il failover per le risorse danneggiate responsabili dell'incremento della latenza dell'elaborazione o che ripetutamente non riescono a elaborare le richieste.
- Utilizza l'elaborazione asincrona. A tale scopo, progetta API in grado di accettare le richieste e aggiungere richieste alle code interne mediante Amazon SQS e, quindi, rispondere al client che genera il messaggio con un messaggio di successo, in modo che il client possa rilasciare risorse e passare ad altre attività mentre gli utenti nella coda di backend elaborano le richieste.
- Misura e monitora la latenza di elaborazione delle code generando una metrica CloudWatch ogni volta che escludi un messaggio da una coda confrontandolo con il timestamp del messaggio.
- Quando gli errori impediscono la corretta elaborazione dei messaggi o il traffico aumenta a livelli tali da impedirne l'elaborazione in base agli accordi sul livello di servizio, escludi il traffico obsoleto o in eccesso indirizzandolo a una coda per il traffico eccedente. Ciò consente l'elaborazione prioritaria del nuovo processo e del processo più vecchio quando si rende disponibile nuova capacità. Questa tecnica è un'approssimazione dell'elaborazione LIFO e consente la normale elaborazione del sistema per tutti i nuovi processi.
- Usa le code DLQ o le code di reindirizzamento per spostare i messaggi che non possono essere elaborati dal backlog in una posizione che può essere ricercata e risolta in un secondo momento.
- Riprova o, se possibile, elimina i vecchi messaggi confrontandoli con il timestamp del messaggio ed eliminando i messaggi che non sono più rilevanti per il client richiedente.

Risorse

Best practice correlate:

- REL04-BP02 Implementare dipendenze liberamente accoppiate
- REL05-BP02 Richieste di limitazione (della larghezza di banda della rete)
- REL05-BP03 Controlla e limita le chiamate di nuovo tentativo
- REL06-BP02 Definizione e calcolo dei parametri (aggregazione)
- REL06-BP07 Monitoraggio del tracciamento end-to-end delle richieste attraverso il sistema

Documenti correlati:

- Evitare insormontabili backlog di code
- Fail Fast

- Come posso prevenire un aumento del backlog di messaggi nella mia coda Amazon SQS?
- Elastic Load Balancing: spostamento zonale
- Amazon Application Recovery Controller: controllo dell'instradamento per il failover del traffico

Esempi correlati:

- Modelli di integrazione aziendale: canale DLQ

Video correlati:

- AWS re:Invent 2022 - Operating highly available Multi-AZ applications

Strumenti correlati:

- Amazon SQS
- Amazon MQ
- AWS IoT Core
- Amazon CloudWatch

REL05-BP05 Imposta i timeout dei client

Imposta i timeout in modo appropriato per connessioni e richieste, verificali sistematicamente e non fare affidamento sui valori predefiniti perché non fanno riferimento alle specifiche del carico di lavoro.

Risultato desiderato: i timeout dei client devono considerare il costo per client, server e carico di lavoro associato all'attesa di richieste il cui completamento richiede una quantità di tempo anomala. Poiché non è possibile conoscere la causa esatta di un timeout, i client devono fare riferimento ai servizi per sviluppare ipotesi sulle cause probabili e sui timeout appropriati.

Il timeout delle connessioni client si verifica in base ai valori configurati. Dopo aver rilevato un timeout, i client decidono di riprovare o aprire un interruttore. Questi modelli evitano la generazione di richieste che potrebbero aggravare una condizione di errore sottostante.

Anti-pattern comuni:

- Non essere a conoscenza dei timeout di sistema o dei timeout predefiniti.
- Non essere a conoscenza dei normali tempi di completamento delle richieste.

- Non essere a conoscenza delle possibili cause dei tempi anomali necessari per il completamento delle richieste o dei costi in termini di prestazioni di client, servizio o carico di lavoro associati all'attesa di tali completamenti.
- Non essere consapevoli della probabilità che una rete danneggiata causi un errore di esecuzione della richiesta solo al raggiungimento del timeout, nonché dei costi in termini di prestazioni del client e del carico di lavoro derivanti dalla mancata adozione di un timeout più breve.
- Non testare gli scenari di timeout sia per le connessioni che per le richieste.
- Impostazione di timeout troppo elevati, che può comportare lunghi tempi di attesa e aumentare l'utilizzo delle risorse.
- Impostazione di timeout troppo bassi, con conseguenti errori artificiali.
- Mancata verifica degli schemi per gestire gli errori di timeout per chiamate remote come interruttori e nuovi tentativi.
- Non considerare il monitoraggio delle percentuali di errore delle chiamate dei servizi, degli obiettivi del livello di servizio per la latenza e dei valori anomali della latenza. Queste metriche possono fornire informazioni sui timeout restrittivi o permissivi.

Vantaggi dell'adozione di questa best practice: i timeout delle chiamate remote sono configurati e i sistemi sono progettati per gestirli correttamente, in modo da preservare le risorse quando le chiamate remote rispondono in modo eccessivamente lento e gli errori di timeout vengono gestiti correttamente dai client di servizio.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Imposta sia un timeout di connessione che un timeout della richiesta su qualsiasi chiamata della dipendenza del servizio e, generalmente, su qualsiasi chiamata tra i processi. Molti framework offrono funzionalità di timeout integrate, ma è necessario prestare attenzione perché alcuni sono caratterizzati da valori predefiniti infiniti o superiori a quelli accettabili per gli obiettivi dei tuoi servizi. Un valore troppo elevato riduce l'utilità del timeout perché le risorse continuano a essere consumate mentre il client attende che si verifichi il timeout. Un valore troppo basso può generare un aumento del traffico sul backend e una maggiore latenza perché vengono ritentate troppe richieste. In alcuni casi, questo può portare a interruzioni vere e proprie perché tutte le richieste vengono ritentate.

Considera quanto segue per determinare le strategie di timeout:

- L'elaborazione delle richieste può richiedere più tempo del normale a causa del loro contenuto, di problemi nel servizio di destinazione o di un errore nella partizione della rete.
- Le richieste con contenuti troppo costosi potrebbero consumare risorse server e client non necessarie. In questo caso, forzare il timeout di queste richieste e non eseguire nuovi tentativi possono preservare le risorse. I servizi dovrebbero, inoltre, proteggersi da contenuti eccessivamente costosi con limitazione (della larghezza di banda della rete) e timeout lato server.
- Per le richieste con tempi di elaborazione eccessivamente lunghi a causa di un'interruzione del servizio è possibile forzare il timeout e, quindi, eseguire un nuovo tentativo. È necessario considerare i costi del servizio per la richiesta e il nuovo tentativo, ma se la causa è un problema localizzato, è probabile che un nuovo tentativo non sia costoso e riduca il consumo di risorse del client. Il timeout può anche liberare risorse del server a seconda della natura del problema.
- Per le richieste il cui completamento richiede troppo tempo o per risposte non distribuite dalla rete è possibile forzare il timeout e, quindi, eseguire un nuovo tentativo. Poiché la richiesta o la risposta non è stata distribuita, viene comunque restituito un errore indipendentemente dalla durata del timeout. Il timeout in questo caso non rilascerà le risorse del server, ma le risorse del client, con il conseguente miglioramento delle prestazioni del carico di lavoro.

Sfrutta gli schemi di progettazione consolidati, come i tentativi e gli interruttori automatici, per gestire i timeout in modo corretto e supportare approcci di tipo fail-fast. AWS SDKse AWS CLIconsentono la configurazione dei timeout di connessione e di richiesta e i nuovi tentativi con backoff e jitter esponenziali. AWS Lambdale funzioni supportano la configurazione dei timeout e, con AWS Step Functions, è possibile creare interruttori automatici a basso codice che sfruttano le integrazioni predefinite con i servizi e. AWS SDKs AWS App Mesh Envoy fornisce funzionalità di tipo timeout e interruttore.

Passaggi dell'implementazione

- Configura i timeout per le chiamate remote dei servizi e sfrutta le funzionalità di timeout integrate o le librerie di timeout open source.
- Quando il tuo carico di lavoro effettua chiamate con un AWS SDK, consulta la documentazione per la configurazione del timeout specifica della lingua.
  - Python
  - PHP
  - .NET
  - Ruby

  - Java
  - Go
  - Node.js
  - C++
- Quando utilizzi AWS CLI i comandi AWS SDKs or nel tuo carico di lavoro, configura i valori di timeout predefiniti impostando i valori di AWS configurazione predefiniti per e. `connectTimeoutInMillis` `tlsNegotiationTimeoutInMillis`
- Applica le opzioni della riga di comando `cli-connect-timeout` e controlla comandi singoli `cli-read-timeout` AWS CLI ai servizi. AWS
- Monitora le chiamate remote dei servizi per i timeout e imposta gli allarmi sugli errori persistenti in modo da poter gestire in modo proattivo gli scenari di errore.
- Implementa le CloudWatch metriche e il rilevamento delle CloudWatch anomalie sui tassi di errore delle chiamate, sugli obiettivi dei livelli di servizio per la latenza e sui valori anomali di latenza per fornire informazioni sulla gestione di timeout eccessivamente aggressivi o permissivi.
- Configura i timeout sulle funzioni Lambda.
- APII client Gateway devono implementare i propri tentativi quando gestiscono i timeout. APIGateway supporta un timeout di integrazione da 50 millisecondi a 29 secondi per le integrazioni downstream e non riprova quando le richieste di integrazione scade.
- Implementa il modello dell'interruttore per evitare di effettuare chiamate remote quando si è verificato il timeout. Apri l'interruttore per evitare chiamate non riuscite e chiudi l'interruttore quando le chiamate rispondono normalmente.
- Per i carichi di lavoro basati su container, esamina le funzionalità di App Mesh Envoy per sfruttare timeout e interruttori integrati.
- Utilizzali AWS Step Functions per creare interruttori automatici a basso codice per chiamate di assistenza remota, in particolare quando richiami integrazioni Step Functions AWS native SDKs e supportate per semplificare il carico di lavoro.

Risorse

Best practice correlate:

- REL05-BP03 Controlla e limita le chiamate di nuovo tentativo
- REL05-BP04 Anticipare l’errore (fail fast) e limitare le code
- REL06-BP07 Monitoraggio del tracciamento end-to-end delle richieste attraverso il sistema

Documenti correlati:

- AWS SDK: Tentativi e timeout
- The Amazon Builders' Library: timeout, nuovi tentativi e backoff con jitter
- Quote e note importanti di Amazon API Gateway
- AWS Command Line Interface: opzioni della riga di comando
- AWS SDK for Java 2.x: Configura API i timeout
- AWS Botocore utilizzando l'oggetto config e Config Reference
- AWS SDK per .NET: nuovi tentativi e timeout
- AWS Lambda: configurazione delle opzioni della funzione Lambda

Esempi correlati:

- Utilizzo del pattern di interruttore automatico con AWS Step Functions e Amazon DynamoDB
- Martin Fowler: CircuitBreaker

Strumenti correlati:

- AWS SDKs
- AWS Lambda
- Amazon SQS
- AWS Step Functions
- AWS Command Line Interface

REL05-BP06 Rendere i sistemi senza stato ove possibile

I sistemi non devono richiedere lo stato né eseguire l'offload dello stato in modo tale che, tra diverse richieste client, non vi sia alcuna dipendenza dai dati archiviati localmente su disco o in memoria. I server possono così essere sostituiti a piacimento senza compromettere la disponibilità.

Quando gli utenti o i servizi interagiscono con un'applicazione, spesso eseguono una serie di interazioni che formano una sessione. Una sessione è un dato univoco per gli utenti che persistono tra le richieste mentre utilizzano l'applicazione. Un'applicazione stateless è un'applicazione che non richiede la conoscenza delle interazioni precedenti e non memorizza le informazioni sulla sessione.

Una volta progettati per essere stateless, è possibile utilizzare servizi di elaborazione serverless, come o. AWS Lambda AWS Fargate

Oltre alla sostituzione dei server, un altro vantaggio delle applicazioni stateless è la possibilità di scalare orizzontalmente perché tutte le risorse di elaborazione disponibili (come EC2 istanze e AWS Lambda funzioni) possono soddisfare qualsiasi richiesta.

Vantaggi dell'adozione di questa best practice: i sistemi con progettazione stateless sono più adattabili alla scalabilità orizzontale, così da poter aggiungere o rimuovere capacità in base alle fluttuazioni di traffico e domanda. Sono inoltre intrinsecamente resistenti ai guasti e offrono flessibilità e agilità allo sviluppo delle applicazioni.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Trasforma le applicazioni in stateless. Le applicazioni stateless consentono la scalabilità orizzontale e sono tolleranti ai guasti di un singolo nodo. Analizza e individua i componenti della tua applicazione che mantengono lo stato dell'architettura. Questo processo ti aiuta a valutare il potenziale impatto della transizione a una progettazione stateless. Un'architettura stateless separa i dati degli utenti ed esegue l'offload dei dati della sessione, offrendo la flessibilità necessaria per scalare ogni componente in modo indipendente al fine di soddisfare le diverse richieste del carico di lavoro e ottimizzare l'utilizzo delle risorse.

Passaggi dell'implementazione

- Individua e comprendi i componenti stateful dell'applicazione.
- Suddividi i dati, separando e gestendo i dati dell'utente dalla logica dell'applicazione principale.
  - Amazon Cognito è in grado di separare i dati degli utenti dal codice dell'applicazione mediante funzionalità come pool di identità, pool di utenti e Amazon Cognito Sync.
  - Puoi separare dati degli utenti con Gestione dei segreti AWS, archiviando i segreti in un luogo sicuro e centralizzato. Il codice dell'applicazione pertanto non dovrà più memorizzare i segreti, rendendolo più sicuro.
  - Per archiviare dati non strutturati di grandi dimensioni, come immagini e documenti, prendi in considerazione l'utilizzo di Amazon S3. L'applicazione può recuperare questi dati quando richiesto, eliminando la necessità di archiviarli in memoria.
  - Utilizza Amazon DynamoDB per archiviare informazioni come i profili utente. L'applicazione può eseguire query su questi dati pressoché in tempo reale.

- Trasferisci i dati della sessione in un database, una cache o in file esterni.
  - Amazon ElastiCache, Amazon DynamoDB, Amazon Elastic File System (Amazon) e EFS Amazon MemoryDB sono esempi AWS di servizi che puoi utilizzare per scaricare i dati della sessione.
- Progetta un'architettura stateless dopo aver identificato lo stato e i dati dell'utente che devono essere mantenuti con la tua soluzione di archiviazione preferita.

Risorse

Best practice correlate:

- REL11-BP03 Automatizza la guarigione su tutti i livelli

Documenti correlati:

- The Amazon Builders' Library: evitare il fallback nei sistemi distribuiti
- The Amazon Builders' Library: evitare insormontabili backlog di code
- The Amazon Builders' Library: sfide e strategie del caching
- Best practice per Stateless Web Tier su AWS

REL05-BP07 Implementare leve di emergenza

Le leve di emergenza sono processi rapidi che possono mitigare l'impatto sulla disponibilità sul carico di lavoro.

Le leve di emergenza disabilitano, applicano la limitazione (della larghezza di banda della rete) o modificano il comportamento di componenti o dipendenze mediante meccanismi noti e testati. Ciò può ridurre i danni causati al carico di lavoro dall'esaurimento delle risorse dovuto ad aumenti imprevisti della domanda e l'impatto dei guasti nei componenti non critici all'interno del carico di lavoro.

Risultato desiderato: l'implementazione delle leve di emergenza consente di definire processi noti e validi per mantenere la disponibilità dei componenti critici nel carico di lavoro. Il carico di lavoro dovrebbe diminuire gradualmente e continuare a svolgere le sue funzioni aziendali critiche durante l'attivazione di una leva di emergenza. Per maggiori dettagli sulla degradazione graduale, vedere REL05-BP01 Implementare graceful degradation per trasformare le dipendenze rigide applicabili in dipendenze morbide.

Anti-pattern comuni:

- L'errore a livello di dipendenze non critiche influisce sulla disponibilità del carico di lavoro principale.
- Mancato test o mancata verifica del comportamento dei componenti critici durante il deterioramento delle prestazioni dei componenti non critici.
- Mancata definizione di criteri chiari e deterministici per l'attivazione o la disattivazione di una leva di emergenza.

Vantaggi dell'adozione di questa best practice: l'implementazione delle leve di emergenza migliora la disponibilità dei componenti critici del carico di lavoro fornendo agli addetti alla risoluzione processi consolidati per rispondere a picchi imprevisti della domanda o a guasti delle dipendenze non critiche.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

- Identifica i componenti critici del tuo carico di lavoro.
- Progetta e definisci l'architettura dei componenti critici del tuo carico di lavoro in modo che sia in grado di sostenere i guasti dei componenti non critici.
- Esegui i test per convalidare il comportamento dei componenti critici in caso di guasti dei componenti non critici.
- Definisci e monitora le metriche o i trigger pertinenti per avviare le procedure relative alle leve di emergenza.
- Definisci le procedure (manuali o automatiche) che includono la leva di emergenza.

Passaggi dell'implementazione

- Identifica i componenti business-critical nel tuo carico di lavoro.
  - Ogni componente tecnico del carico di lavoro deve essere mappato alla funzione aziendale pertinente e classificato come critico o non critico. Per esempi di funzionalità critiche e non critiche di Amazon, consulta Any Day Can Be Prime Day: How Amazon.com Search Uses Chaos Engineering to Handle Over 84K Requests Per Second.
  - Si tratta di una decisione sia tecnica che aziendale e varia in base all'organizzazione e al carico di lavoro.

- Progetta e definisci l'architettura dei componenti critici del tuo carico di lavoro in modo che sia in grado di sostenere i guasti dei componenti non critici.
  - Durante l'analisi delle dipendenze, valuta tutte le potenziali modalità di guasto e verifica che i meccanismi basati su leve di emergenza forniscano le funzionalità critiche ai componenti a valle.
- Esegui i test per convalidare il comportamento dei componenti critici durante l'attivazione delle leve di emergenza.
  - Evita il comportamento bimodale. Per maggiori dettagli, consulta -BP05 Utilizzare la stabilità statica per prevenire comportamenti bimodali. REL11
- Definisci, monitora e attiva gli avvisi per le metriche pertinenti per avviare la procedura relative alla leva di emergenza.
  - L'individuazione delle metriche da monitorare dipende dal carico di lavoro. Alcuni esempi di metrica sono la latenza o il numero di richieste non riuscite nei confronti di una dipendenza.
- Definisci le procedure (manuali o automatiche) che includono la leva di emergenza.
  - Ciò può includere meccanismi come la riduzione del carico, le richieste di limitazione (della larghezza di banda della rete) o l'implementazione della normale riduzione delle prestazioni.

Risorse

Best practice correlate:

- REL05-BP01 Implementa una degradazione graduale per trasformare le dipendenze rigide applicabili in dipendenze morbide
- REL05-BP02 Richieste Throttle
- REL11-BP05 Usa la stabilità statica per prevenire comportamenti bimodali

Documenti correlati:

- Automatizzazione di distribuzioni pratiche e sicure
- Any Day Can Be Prime Day: How Amazon.com Search Uses Chaos Engineering to Handle Over 84K Requests Per Second

Video correlati:

- AWS re:Invent 2020: affidabilità, coerenza e fiducia grazie all'immutabilità

Gestione delle modifiche

Questions

- REL 6. Come si monitorano le risorse del carico di lavoro?
- REL 7. Come si progetta il carico di lavoro per adattarsi ai cambiamenti della domanda?
- REL 8. In che modo implementare le modifiche?

REL 6. Come si monitorano le risorse del carico di lavoro?

I log e le metriche sono strumenti molto efficaci per ottenere informazioni sullo stato del carico di lavoro. Puoi configurare il carico di lavoro in modo da monitorare i log e le metriche e inviare notifiche in caso di superamento delle soglie o di eventi significativi. Il monitoraggio permette al carico di lavoro di riconoscere il superamento delle soglie di prestazioni basse o il verificarsi di errori, in modo da ripristinarlo in automatico di conseguenza.

Best practice

- REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)
- REL06-BP02 Definizione e calcolo dei parametri (aggregazione)
- REL06-BP03 Invio di notifiche (elaborazione e avvisi in tempo reale)
- REL06-BP04 Automatizzazione delle risposte (elaborazione e avvisi in tempo reale)
- REL06-BP05 Analisi dei log
- REL06-BP06 Revisione periodica dell'ambito e delle metriche di monitoraggio
- REL06-BP07 Monitoraggio del tracciamento end-to-end delle richieste attraverso il sistema

REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)

Monitora i componenti del carico di lavoro con Amazon CloudWatch o con strumenti di terze parti. Monitora i servizi AWS con il pannello di controllo AWS Health.

Occorre monitorare tutti i componenti del carico di lavoro, inclusi front-end, logica aziendale e livelli di storage. Definisci i parametri chiave e come estrarli dai log, se necessario, e imposta soglie per richiamare gli eventi di allarme corrispondenti. Assicurati che i parametri siano pertinenti agli indicatori chiave di prestazione (KPI) del tuo carico di lavoro e utilizza i parametri e i log per identificare i primi segnali di degrado del servizio. Ad esempio, un parametro legato ai risultati aziendali, come il numero

di ordini elaborati con successo al minuto, può indicare problemi di carico di lavoro più rapidamente di un parametro tecnico, come l'utilizzo della CPU. Utilizza il pannello di controllo AWS Health per una visualizzazione personalizzata delle prestazioni e della disponibilità dei servizi AWS sottostanti alle risorse AWS.

Il monitoraggio nel cloud offre nuove opportunità. La maggior parte dei provider cloud ha sviluppato hook personalizzabili e può fornire approfondimenti per aiutarti a monitorare più livelli del carico di lavoro. I servizi AWS come Amazon CloudWatch applicano algoritmi statistici e di machine learning per analizzare continuamente i parametri di sistemi e applicazioni, determinare le normali linee di base e far emergere le anomalie con un intervento minimo da parte dell'utente. Gli algoritmi di rilevamento delle anomalie tengono conto delle variazioni di stagionalità e di tendenza dei parametri.

AWS mette a disposizione una grande quantità di informazioni di monitoraggio e di log che possono essere utilizzate per definire parametri specifici per i carichi di lavoro, processi di variazione della domanda e per l'adozione di tecniche di machine learning indipendentemente dalle competenze di ML.

Inoltre, monitora tutti gli endpoint esterni per avere la certezza che siano indipendenti dall'implementazione di base. Questo monitoraggio attivo può essere svolto attraverso transazioni sintetiche (talvolta definite canary dell'utente, ma da non confondere con le distribuzioni canary) che eseguono periodicamente alcune attività comuni che corrispondono a quelle effettuate dai client del carico di lavoro. Mantieni queste attività di breve durata e assicurati di non sovraccaricare il carico di lavoro durante il test. Amazon CloudWatch Synthetics consente di creare canary sintetici per monitorare endpoint e API. Puoi anche combinare i nodi client sintetici canary con la console AWS X-Ray per individuare quali canary sintetici stanno riscontrando problemi con errori, guasti o tassi di limitazione (della larghezza di banda della rete) per l'intervallo di tempo selezionato.

Risultato desiderato:

Raccogliere e utilizzare i parametri critici di tutti i componenti del carico di lavoro per garantire l'affidabilità del carico di lavoro e un'esperienza utente ottimale. Rilevare che un carico di lavoro non sta raggiungendo i risultati aziendali consente di dichiarare rapidamente un disastro e di riprendersi da un incidente.

Anti-pattern comuni:

- Solo monitoraggio delle interfacce esterne per il carico di lavoro.
- Non generare parametri specifici per il carico di lavoro e affidati solo ai parametri forniti dai servizi AWS utilizzati dal carico di lavoro.

- Utilizzare solo parametri tecnici nel carico di lavoro e non monitorare i parametri relativi agli indicatori chiave di prestazione (KPI) non tecnici a cui il carico di lavoro contribuisce.
- Affidarsi al traffico di produzione e a semplici controlli dell'integrità per monitorare e valutare lo stato del carico di lavoro.

Vantaggi dell'adozione di questa best practice: il monitoraggio a tutti i livelli del carico di lavoro consente di prevedere e risolvere più rapidamente i problemi dei componenti che costituiscono il carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

1. Attiva la creazione di log, laddove possibile. I dati di monitoraggio devono essere ottenuti da tutti i componenti dei carichi di lavoro. Attiva ulteriori log, come i log di accesso S3, e consenti al carico di lavoro di creare log per i dati specifici del carico di lavoro. Raccogli i parametri relativi a CPU, I/O di rete e I/O su disco da servizi come Amazon ECS, Amazon EKS, Amazon EC2, Elastic Load Balancing, AWS Auto Scaling, e Amazon EMR. Consulta AWS Services That Publish CloudWatch Metrics per un elenco di servizi AWS che pubblicano parametri su CloudWatch.
2. Esamina tutti i parametri predefiniti ed esplora eventuali lacune nella raccolta dei dati. Tutti i servizi generano parametri predefiniti. La raccolta di parametri predefiniti consente di comprendere meglio le dipendenze tra i componenti del carico di lavoro e il modo in cui l'affidabilità e le prestazioni dei componenti influiscono sul carico di lavoro. Puoi anche creare e pubblicare i tuoi parametri in CloudWatch tramite la AWS CLI o un'API.
3. Valuta tutti i parametri per decidere quelli a cui inviare avvisi per ogni servizio AWS nel carico di lavoro. Puoi scegliere di selezionare un sottoinsieme di parametri che hanno un impatto importante sull'affidabilità del carico di lavoro. Concentrarsi su parametri e soglie critiche consente di affinare il numero di avvisi, così da ridurre al minimo i falsi positivi.
4. Definisci gli avvisi e il processo di recupero del carico di lavoro dopo il richiamo dell'avviso. La definizione degli avvisi consente di notificare, intensificare e seguire rapidamente le fasi necessarie per il ripristino da un incidente e il rispetto dell'Obiettivo del tempo di ripristino (RTO). Puoi usare gli allarmi di Amazon CloudWatch per richiamare flussi di lavoro automatici e avviare procedure di ripristino in base a soglie definite.
5. Esplora l'uso di transazioni sintetiche per raccogliere dati rilevanti sullo stato dei carichi di lavoro. Il monitoraggio sintetico segue gli stessi percorsi ed esegue le stesse azioni di un cliente, il che consente di verificare continuamente l'esperienza del cliente anche quando non c'è traffico di

clienti sui carichi di lavoro. Grazie alle transazioni sintetiche, puoi scoprire i problemi prima che vengano rilevati dai clienti.

Risorse

Best practice correlate:

- REL11-BP03 Automatizzazione della riparazione a tutti i livelli

Documenti correlati:

- Getting started with your AWS Health Dashboard – Your account health
- AWS Services That Publish CloudWatch Metrics
- Access Logs for Your Network Load Balancer
- Access logs for your application load balancer
- Accesso ad Amazon CloudWatch Logs per AWS Lambda
- Registrazione degli accessi al server Amazon S
- Enable Access Logs for Your Classic Load Balancer
- Exporting log data to Amazon S3
- Install the CloudWatch agent on an Amazon EC2 instance
- Publishing Custom Metrics
- Using Amazon CloudWatch Dashboards
- Using Amazon CloudWatch Metrics
- Utilizzo di Canary (Amazon CloudWatch Synthetics)
- What are Amazon CloudWatch Logs?

Guide per l'utente:

- Creating a trail
- Monitoraggio dei parametri relativi a memoria e disco per le istanze Linux di Amazon EC2
- Utilizzo dei CloudWatch Logs con le istanze di container
- Log di flusso VPC
- What is Amazon DevOps Guru?
- Cos'è AWS X-Ray?

Blog correlati:

- Debugging with Amazon CloudWatch Synthetics and AWS X-Ray

Esempi correlati:

- The Amazon Builders' Library: strumentazione di sistemi distribuiti per visibilità operativa
- Workshop sull'osservabilità

REL06-BP02 Definizione e calcolo dei parametri (aggregazione)

Raccogli metriche e log dai componenti del carico di lavoro e calcola le metriche aggregate pertinenti. Queste metriche forniscono un'ampia e profonda osservabilità del carico di lavoro e possono migliorare in modo significativo lo stato di resilienza.

L'osservabilità non si limita alla semplice raccolta di metriche dai componenti del carico di lavoro e alla possibilità di visualizzarle e di emettere avvisi. Si tratta di avere una comprensione olistica del comportamento del carico di lavoro. Queste informazioni comportamentali provengono da tutti i componenti dei carichi di lavoro, compresi i servizi cloud da cui dipendono, i log ben realizzati e le metriche. Questi dati consentono di controllare il comportamento del carico di lavoro nel suo complesso e di comprendere l'interazione di ogni componente con ogni unità di lavoro a un livello di dettaglio molto elevato.

Risultato desiderato:

- Raccogli i log dai componenti del carico di lavoro e dalle dipendenze dei servizi AWS e li pubblichi in una posizione centrale dove possono essere facilmente consultati ed elaborati.
- I log contengono timestamp accurati e ad alta fedeltà.
- I log contengono informazioni rilevanti sul contesto di elaborazione, come l'identificativo della traccia, l'identificativo dell'utente o dell'account e l'indirizzo IP remoto.
- Dai log crei metriche aggregate che rappresentano il comportamento del carico di lavoro da una prospettiva di alto livello.
- Puoi eseguire query sui log aggregati per ottenere informazioni approfondite e pertinenti sul carico di lavoro e identificare problemi effettivi e potenziali.

Anti-pattern comuni:

- Non raccogli log o metriche pertinenti dalle istanze di calcolo su cui vengono eseguiti i carichi di lavoro o dai servizi cloud che utilizzano.
- Trascuri la raccolta di log e metriche collegate agli indicatori chiave delle prestazioni (KPI) aziendali.
- Analizzi la telemetria correlata al carico di lavoro in modo isolato, senza aggregazione e correlazione.
- Consenti che le metriche e i log scadano troppo rapidamente, il che ostacola l'analisi delle tendenze e l'identificazione dei problemi ricorrenti.

Vantaggi dell'adozione delle best practice: puoi rilevare un maggior numero di anomalie e correlare eventi e metriche tra i diversi componenti del carico di lavoro. Puoi creare approfondimenti sui componenti del carico di lavoro in base alle informazioni contenute nei log che spesso non sono disponibili nelle sole metriche. Puoi determinare più rapidamente le cause degli errori eseguendo query sui log su larga scala.

Livello di rischio associato se queste best practice non fossero adottate: elevato

Guida all'implementazione

Identifica le origini di dati di telemetria rilevanti per i carichi di lavoro e i relativi componenti. Questi dati provengono non solo dai componenti che pubblicano metriche, come il sistema operativo (OS) e i runtime delle applicazioni come Java, ma anche dai log delle applicazioni e dei servizi cloud. Ad esempio, i server web in genere registrano ogni richiesta con informazioni dettagliate come il timestamp, la latenza di elaborazione, l'ID utente, l'indirizzo IP remoto, il percorso e la stringa di query. Il livello di dettaglio di questi log consente di eseguire query dettagliate e di generare metriche che altrimenti non sarebbero disponibili.

Raccogli le metriche e i log utilizzando strumenti e processi appropriati. I log generati dalle applicazioni in esecuzione su un'istanza Amazon EC2 possono essere raccolti da un agente come Agente Amazon CloudWatch e pubblicati su un servizio di archiviazione centrale come Amazon CloudWatch Logs. I servizi di elaborazione gestiti da AWS come AWS Lambda e Amazon Elastic Container Service pubblicano automaticamente i log su CloudWatch Logs. Abilita la raccolta di log per i servizi di archiviazione ed elaborazione AWS utilizzati dai carichi di lavoro come Amazon CloudFront, Amazon S3, Elastic Load Balancing e Gateway Amazon API.

Arricchisci i dati di telemetria con dimensioni che possono aiutarti a vedere più chiaramente i modelli di comportamento e a isolare i problemi relativi a gruppi di componenti correlati. Una volta aggiunte, è

possibile osservare il comportamento dei componenti a un livello di dettaglio più fine, rilevare gli errori correlati e adottare le operazioni correttive appropriate. Esempi di dimensioni utili includono zona di disponibilità, ID istanza EC2 e attività del container o Pod ID.

Dopo aver raccolto le metriche e i log, puoi scrivere query e generare metriche aggregate che forniscono informazioni utili sul comportamento normale e anomalo. Ad esempio, puoi utilizzare Approfondimenti di Amazon CloudWatch Logs per ricavare parametri personalizzati dai log delle applicazioni, approfondimenti sulle metriche Amazon CloudWatch per eseguire query sui parametri su larga scala, approfondimenti sui container Amazon CloudWatch per raccogliere, aggregare e riepilogare metriche e log dalle applicazioni e microservizi containerizzati o Lambda Insights di Amazon CloudWatch se utilizzi le funzioni AWS Lambda. Per creare una metrica aggregata del tasso di errore, è possibile incrementare un contatore ogni volta che si trova una risposta o un messaggio di errore nei log del componente o calcolare il valore aggregato di una metrica del tasso di errore esistente. Puoi utilizzare questi dati per generare istogrammi che mostrano il comportamento della coda, come le richieste o i processi con le prestazioni peggiori. Puoi anche eseguire la scansione di questi dati in tempo reale alla ricerca di modelli anomali utilizzando soluzioni come il rilevamento delle anomalie di CloudWatch Logs. Queste informazioni approfondite possono essere inserite in dashboard per essere organizzate in base alle esigenze e alle preferenze.

L'esecuzione di query sui log può aiutare a comprendere come sono state gestite richieste specifiche dai componenti del carico di lavoro e a rivelare modelli di richiesta o altri contesti che hanno un impatto sulla resilienza del carico di lavoro. Può essere utile ricercare e preparare le query in anticipo, in base alla conoscenza del comportamento delle applicazioni e degli altri componenti, in modo da poterle eseguire più facilmente quando necessario. Ad esempio, con Approfondimenti di Amazon CloudWatch Logs, puoi cercare e analizzare in modo interattivo i dati di log memorizzati in CloudWatch Logs. Puoi anche usare Amazon Athena per eseguire query sui log da più origini, inclusi molti servizi AWS, su una scala di petabyte.

Quando si definisce una policy di conservazione dei log, devi considerare il valore dei log storici. I log storici possono aiutare a identificare modelli di utilizzo e di comportamento a lungo termine, regressioni e miglioramenti delle prestazioni del carico di lavoro. I log eliminati definitivamente non possono essere analizzati in seguito. Tuttavia, il valore dei log storici tende a diminuire nel corso di lunghi periodi di tempo. Scegli una policy che sia in grado di bilanciare le esigenze e che sia conforme ai requisiti legali o contrattuali a cui potresti essere soggetto.

Passaggi dell'implementazione

1. Scegli i meccanismi di raccolta, archiviazione, analisi e visualizzazione dei dati di osservabilità.

2. Installa e configura i raccoglitori di metriche e di log sui componenti appropriati del carico di lavoro (ad esempio, sulle istanze Amazon EC2 e nei container sidecar). Configura questi raccoglitori in modo che si riavviino automaticamente nel caso in cui si arrestino in modo imprevisto. Abilita il buffering su disco o in memoria per i collettori, in modo che le interruzioni temporanee della pubblicazione non abbiano ripercussioni sulle applicazioni né comportino la perdita di dati.
3. Abilita la registrazione sui servizi AWS utilizzati come parte dei carichi di lavoro e inoltra i log al servizio di archiviazione selezionato, se necessario. Per istruzioni dettagliate, consulta le guide per l'utente o gli sviluppatori dei rispettivi servizi.
4. Definisci le metriche operative rilevanti per i carichi di lavoro, basate sui dati di telemetria. Questi possono essere basati su metriche dirette emesse dai componenti del carico di lavoro, che possono includere metriche correlate a KPI aziendali, o sui risultati di calcoli aggregati come somme, tassi, percentili o istogrammi. Calcola queste metriche utilizzando l'analizzatore log e inseriscile nelle dashboard come opportuno.
5. Prepara query di log appropriate per analizzare i componenti del carico di lavoro, le richieste o il comportamento delle transazioni, se necessario.
6. Definisci e abilita una policy di conservazione dei log per i log dei componenti. Elimina periodicamente i log quando diventano più vecchi di quanto consentito dalla policy.

Risorse

Best practice correlate:

- REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)
- REL06-BP03 Invio di notifiche (elaborazione e avvisi in tempo reale)
- REL06-BP04 Automatizzazione delle risposte (elaborazione e avvisi in tempo reale)
- REL06-BP05 Analisi dei log
- REL06-BP06 Revisione periodica dell'ambito e delle metriche di monitoraggio
- REL06-BP07 Monitoraggio del tracciamento end-to-end delle richieste attraverso il sistema

Documentazione correlata:

- How Amazon CloudWatch works
- Amazon Managed Prometheus
- Grafana gestito da Amazon

- Analyzing log data with CloudWatch Logs Insights
- Lambda Insights di Amazon CloudWatch
- Approfondimenti sui container Amazon CloudWatch
- Query your metrics with CloudWatch Metrics Insights
- AWS Distro per OpenTelemetry
- Query di esempio di Approfondimenti di Amazon CloudWatch Logs
- Debugging with Amazon CloudWatch Synthetics and AWS X-Ray
- Ricerca e filtraggio dei dati di log
- Invio di log direttamente ad Amazon S3
- The Amazon Builders' Library: strumentazione di sistemi distribuiti per visibilità operativa

Workshop correlati:

- One Observability Workshop

Strumenti correlati:

- AWS Distro for OpenTelemetry (GitHub)

REL06-BP03 Invio di notifiche (elaborazione e avvisi in tempo reale)

Quando le organizzazioni rilevano potenziali problemi, inviano notifiche e avvisi in tempo reale ai team e ai sistemi appropriati per rispondere rapidamente ed efficacemente alle difficoltà.

Risultato desiderato: è possibile rispondere rapidamente agli eventi operativi attraverso la configurazione di allarmi pertinenti in base ai parametri del servizio e dell'applicazione. Quando la soglia degli allarmi viene superata, i team e i sistemi appropriati vengono informati in modo che possano risolvere i problemi sottostanti.

Anti-pattern comuni:

- Configuri gli allarmi con una soglia eccessivamente alta, con conseguente mancato invio di notifiche importanti.
- Configuri gli allarmi con una soglia troppo bassa, con il risultato che gli avvisi importanti non vengono presi in considerazione a causa del numero eccessivo di notifiche generate.
- Non aggiorni gli allarmi e la relativa soglia quando cambia l'utilizzo.

- Per gli allarmi gestiti meglio tramite le azioni automatizzate, l'invio della notifica ai team anziché l'attivazione dell'azione automatizzata comporta la generazione di un numero eccessivo di notifiche.

Vantaggi dell'adozione di questa best practice: l'invio di notifiche e avvisi in tempo reale ai team e ai sistemi appropriati consente di individuare tempestivamente i problemi e di rispondere rapidamente agli incidenti operativi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I carichi di lavoro devono essere dotati di sistemi di elaborazione e allarme in tempo reale per migliorare l'identificazione dei problemi che possono influire sulla disponibilità dell'applicazione e fungere da trigger per la risposta automatizzata. Le organizzazioni possono eseguire un sistema di elaborazione e allarme in tempo reale creando avvisi con parametri definiti in modo da ricevere le notifiche ogni volta che si verificano eventi significativi o un parametro supera una determinata soglia.

Amazon CloudWatch consente di creare allarmi di parametri e compositi mediante gli allarmi CloudWatch basati su soglie statiche, rilevamento di anomalie e altri criteri. Per ulteriori informazioni sui tipi di allarmi configurabili mediante CloudWatch, consulta la sezione sugli allarmi della documentazione di CloudWatch.

Puoi creare per i tuoi team visualizzazioni personalizzate dei parametri e degli avvisi delle risorse AWS utilizzando i pannelli di controllo di CloudWatch. Le home page personalizzabili nella console di CloudWatch consentono di monitorare le risorse di più regioni in un'unica visualizzazione.

Gli allarmi possono eseguire una o più azioni, come inviare una notifica a un argomento Amazon SNS, eseguendo un'azione Amazon EC2 o un'azione Amazon EC2 Auto Scaling oppure creando un OpsItem o un incidente in AWS Systems Manager.

Amazon CloudWatch utilizza Amazon SNS per inviare le notifiche quando l'allarme cambia stato, con la distribuzione dei messaggi degli editori (produttori) agli abbonati (consumatori). Per ulteriori informazioni sulla configurazione delle notifiche di Amazon SNS, consulta Configuring Amazon SNS.

CloudWatch invia eventi EventBridge ogni volta che un allarme CloudWatch viene creato, aggiornato, eliminato o cambia stato. Puoi usare EventBridge con questi eventi per creare le regole che eseguono le azioni, come avvisare ogni volta che lo stato di un allarme cambia o attivare automaticamente gli eventi nel tuo account tramite l'automazione di Systems Manager.

Con AWS Health si rimane sempre aggiornati. AWS Health è la fonte autorevole di informazioni sull'integrità delle risorse Cloud AWS. AWS Health consente di ricevere le notifiche in caso di eventi del servizio confermati, in modo da poter adottare rapidamente le misure necessarie per mitigare qualsiasi impatto. Si creano notifiche di eventi AWS Health personalizzati per i canali e-mail e chat con Notifiche all'utente AWS e si usano gli strumenti di monitoraggio e avviso con Amazon EventBridge per l'integrazione a livello di codice. Se si utilizza AWS Organizations, è possibile aggregare gli eventi AWS Health tra gli account.

Quando utilizzare EventBridge o Amazon SNS?

EventBridge e Amazon SNS possono entrambi essere utilizzati per sviluppare applicazioni basate su eventi e la scelta dipende dalle tue esigenze specifiche.

Amazon EventBridge è consigliato per creare applicazioni che reagiscano agli eventi delle applicazioni, delle applicazioni SaaS e dei servizi AWS. EventBridge è l'unico servizio basato su eventi integrato direttamente con partner SaaS di terze parti. EventBridge inoltre acquisisce automaticamente eventi da oltre 200 servizi AWS senza che gli sviluppatori debbano creare risorse negli account.

EventBridge utilizza una struttura definita basata su JSON per gli eventi e consente di creare regole applicate all'intero corpo dell'evento per selezionare gli eventi da inoltrare alle destinazioni. EventBridge supporta al momento oltre 20 servizi AWS come destinazione, tra cui AWS Lambda, Amazon SQS, Amazon SNS, flusso di dati Amazon Kinesis e Amazon Data Firehose.

Amazon SNS è consigliato per le applicazioni che richiedono un fan-out elevato (migliaia o milioni di endpoint). Di solito i clienti utilizzano Amazon SNS come destinazione della regola per filtrare gli eventi di cui hanno bisogno e sottoporli al fan-out su più endpoint.

I messaggi non sono strutturati e possono assumere qualsiasi formato. Amazon SNS consente di inoltrare messaggi a sei diversi tipi di destinazioni, tra cui Lambda, Amazon SQS, endpoint HTTP/S, SMS, push mobile ed e-mail. La latenza tipica di Amazon SNS è inferiore a 30 millisecondi. Un'ampia gamma di servizi AWS invia i messaggi Amazon SNS definendo la configurazione appropriata (più di 30, inclusi, Amazon EC2, Amazon S3 e Amazon RDS).

Passaggi dell'implementazione

1. Crea un allarme mediante gli allarmi di Amazon CloudWatch.
    a. Un allarme di parametri monitora un singolo parametro CloudWatch o un'espressione dipendente dai parametri CloudWatch. L'allarme avvia una o più azioni in base al valore del

parametro o dell'espressione rispetto a una soglia, per un determinato numero di intervalli di tempo. L'azione può consistere nell'inviare una notifica a un argomento Amazon SNS, nell'esecuzione di un'azione Amazon EC2 o un'azione Amazon EC2 Auto Scaling oppure nella creazione di un OpsItem o di un incidente in AWS Systems Manager.

b. Un allarme composito è costituito da un'espressione di regola che considera le condizioni di altri allarmi che hai creato. L'allarme composito entra in stato di allarme solo se tutte le condizioni della regola sono soddisfatte. Gli allarmi specificati nell'espressione di regola di un allarme composito possono includere allarmi di parametri e allarmi compositi aggiuntivi. Gli allarmi compositi possono inviare notifiche di Amazon SNS quando cambiano stato e possono creare oggetti OpsItem di Systems Manager o incidenti quando passano allo stato di allarme, ma non possono eseguire azioni EC2 o azioni Auto Scaling.

2. Configura le notifiche di Amazon SNS. Quando si crea un allarme CloudWatch, è possibile includere un argomento Amazon SNS per inviare una notifica quando l'allarme cambia stato.
3. Crea regole in EventBridge corrispondenti agli allarmi CloudWatch specificati. Ogni regola supporta più destinazioni, incluse le funzioni Lambda. Ad esempio, è possibile definire un allarme che si attiva quando lo spazio disponibile su disco si sta esaurendo, il che attiva una funzione Lambda tramite una regola EventBridge per ripulire lo spazio. Per ulteriori informazioni sulle destinazioni EventBridge, consulta EventBridge targets.

Risorse

Best practice Well-Architected correlate:

- REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)
- REL06-BP02 Definizione e calcolo dei parametri (aggregazione)
- REL12-BP01 Utilizzo dei playbook per analizzare gli errori

Documenti correlati:

- Amazon CloudWatch
- CloudWatch Logs insights
- Using Amazon CloudWatch alarms
- Using Amazon CloudWatch dashboards
- Using Amazon CloudWatch metrics
- Impostazione delle notifiche Amazon SNS

- CloudWatch anomaly detection
- Protezione dei dati di CloudWatch Logs
- Amazon EventBridge
- Amazon Simple Notification Service

Video correlati:

- Video sull'osservabilità reinvent
- AWS re:Invent 2022 - Observability best practices at Amazon

Esempi correlati:

- One Observability Workshop
- Amazon EventBridge to AWS Lambda with feedback control by Amazon CloudWatch Alarms

REL06-BP04 Automatizzazione delle risposte (elaborazione e avvisi in tempo reale)

Utilizza l'automazione per agire quando viene rilevato un evento; ad esempio, per sostituire i componenti guasti.

L'elaborazione automatizzata in tempo reale degli allarmi è implementata in modo che i sistemi possano effettuare azioni correttive rapide e tentare di prevenire guasti o danni al servizio quando vengono attivati gli allarmi. Le risposte automatiche agli allarmi potrebbero includere la sostituzione dei componenti guasti, la regolazione della capacità di calcolo, il reindirizzamento del traffico verso host integri, zone di disponibilità o altre regioni e la notifica agli operatori.

Risultato desiderato: identificazione degli allarmi in tempo reale e impostazione dell'elaborazione automatica degli allarmi per richiamare le azioni appropriate intraprese per rispettare gli obiettivi dei livelli di servizio e gli accordi sul livello di servizio (SLA) L'automazione può interessare un ambito che va dalle attività di autoriparazione dei singoli componenti al failover dell'intero sito.

Anti-pattern comuni:

- Non disporre di un inventario o un catalogo dettagliato dei principali allarmi in tempo reale.
- Nessuna risposta automatica in caso di allarmi critici (ad esempio, quando le risorse di calcolo stanno per esaurirsi, viene implementato il dimensionamento automatico).

- Azioni di risposta agli allarmi contraddittorie.
- Nessuna procedura operativa standard (SOP) da seguire per gli operatori quando ricevono notifiche di avviso.
- Non monitorare le modifiche apportate alla configurazione, poiché le modifiche della configurazione non rilevate possono causare tempi di inattività per i carichi di lavoro.
- Non avere una strategia per annullare le modifiche involontarie alla configurazione.

Vantaggi dell'adozione di questa best practice: migliore resilienza del sistema grazie all'automazione dell'elaborazione degli allarmi. Il sistema implementa automaticamente azioni correttive, riducendo le attività manuali che possono comportare interventi umani soggetti a errori. L'operatività del carico di lavoro soddisfa gli obiettivi di disponibilità e riduce le interruzioni del servizio.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Per gestire in modo efficiente gli avvisi e automatizzarne la risposta, classifica gli avvisi in base alla loro criticità e al loro impatto, documenta le procedure di risposta e pianifica le risposte prima di classificare le attività.

Identifica le attività che richiedono azioni specifiche (spesso dettagliate nei runbook) ed esamina tutti i runbook e i playbook per determinare quali attività possono essere automatizzate. Se è possibile definire delle azioni, significa che esse spesso possono essere automatizzate. Se le azioni non possono essere automatizzate, documenta le fasi manuali in una procedura operativa standard (SOP) e forma gli operatori su tali procedure. Continua ad analizzare dettagliatamente i processi manuali alla ricerca di opportunità di automazione in cui puoi stabilire e mantenere un piano per automatizzare le risposte agli avvisi.

Passaggi dell'implementazione

1. Crea un inventario degli allarmi: per ottenere un elenco di tutti gli allarmi, utilizza AWS CLI mediante il comando `describe-alarms` di Amazon CloudWatch. In base al numero di allarmi impostati, potrebbe essere necessario utilizzare la paginazione per recuperare un sottoinsieme di allarmi per ciascuna chiamata o, in alternativa, è possibile utilizzare l'SDK AWS per recuperare gli allarmi utilizzando una chiamata API.
2. Documenta tutte le azioni associate all'allarme: aggiorna un runbook con tutti gli allarmi e le relative azioni, a prescindere che siano manuali o automatizzati. AWS Systems Manager offre runbook predefiniti. Per informazioni sull'uso dei runbook, consulta Working with runbooks. Per

informazioni sulla visualizzazione dei contenuti dei runbook, consulta Visualizza il contenuto del runbook.

3. Configura e gestisci le azioni associate all'allarme: per tutti gli allarmi che richiedono un'azione, specifica l'azione automatizzata mediante l'SDK CloudWatch. Ad esempio, puoi modificare automaticamente lo stato delle tue istanze Amazon EC2 in base a un allarme CloudWatch creando e abilitando o disabilitando le azioni associate a un allarme.

   Puoi utilizzare Amazon EventBridge per rispondere automaticamente agli eventi di sistema, come i problemi relativi alla disponibilità delle applicazioni o le modifiche delle risorse. Puoi creare regole che indichino a quali eventi sei interessato e quali operazioni automatizzate eseguire quando un evento corrisponde a una regola. Le azioni avviabili in automatico includono il richiamare una funzione AWS Lambda, il richiamare `Run Command` di Amazon EC2, l'inoltro dell'evento al flusso di dati Amazon Kinesis e la visualizzazione del comando Automate di Amazon EC2 mediante EventBridge.
4. Procedure operative standard (SOP): in base ai componenti dell'applicazione, AWS Resilience Hub suggerisce più modelli SOP. È possibile utilizzare queste SOP per documentare tutti i processi che un operatore deve seguire nel caso in cui venga generato un avviso. È altresì possibile creare una SOP in base alle raccomandazioni di Resilience Hub, laddove sia necessaria un'applicazione Resilience Hub con una policy di resilienza associata, nonché valutare a livello cronologico la resilienza rispetto a tale applicazione. Le raccomandazioni per la SOP sono prodotte dalla valutazione della resilienza.

   Resilience Hub lavora in sinergia con Systems Manager per automatizzare le fasi delle SOP, fornendo una serie di documenti SSM utilizzabili come base per tali SOP. Ad esempio, Resilience Hub può consigliare una SOP per aggiungere spazio su disco in base a un documento SSM di automazione esistente.
5. Esegui azioni automatizzate utilizzando Amazon DevOps Guru: puoi usare Amazon DevOps Guru per monitorare automaticamente le risorse dell'applicazione al fine di rilevare comportamenti anomali e fornire raccomandazioni mirate per accelerare i tempi di identificazione e riparazione dei problemi. DevOps Guru consente di monitorare flussi di dati operativi quasi in tempo reale da più origini, tra cui i parametri di Amazon CloudWatch, AWS Config, AWS CloudFormation e AWS X-Ray. Inoltre, puoi usare DevOps Guru per creare in automatico OpsItems in OpsCenter e inviare eventi a EventBridge per un ulteriore livello di automazione.

Risorse

Best practice correlate:

- REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)
- REL06-BP02 Definizione e calcolo dei parametri (aggregazione)
- REL06-BP03 Invio di notifiche (elaborazione e avvisi in tempo reale)
- REL08-BP01 Utilizzo di runbook per attività standard come l'implementazione

Documenti correlati:

- AWS Systems Manager Automation
- Creazione di una regola EventBridge che attivi un evento da una risorsa AWS
- One Observability Workshop
- The Amazon Builders' Library: strumentazione di sistemi distribuiti per visibilità operativa
- What is Amazon DevOps Guru?
- Utilizzo dei documenti di automazione (playbook)

Video correlati:

- AWS re:Invent 2022 - Observability best practices at Amazon
- AWS re:Invent 2020: Automate anything with AWS Systems Manager
- Introduction to AWS Resilience Hub
- Create Custom Ticket Systems for Amazon DevOps Guru Notifications
- Enable Multi-Account Insight Aggregation with Amazon DevOps Guru

Esempi correlati:

- Workshop su Amazon CloudWatch e Systems Manager

REL06-BP05 Analisi dei log

Raccogli i file di log e le cronologie dei parametri e analizzali per ottenere informazioni più ampie sulle tendenze e sui carichi di lavoro.

Gli Approfondimenti di Amazon CloudWatch Logs supportano un linguaggio di query semplice ma potente che puoi utilizzare per analizzare i dati di log. Amazon CloudWatch Logs supporta anche le sottoscrizioni che consentono il flusso di dati in modo ottimale verso Amazon S3, dove è possibile utilizzare Amazon Athena per eseguire query sui dati. Supporta, inoltre, le query su un'ampia gamma

di formati. Consulta Supported SerDes and Data Formats nella Guida per l'utente di Amazon Athena. Per l'analisi di enormi set di file di log, è possibile eseguire un cluster Amazon EMR per eseguire analisi con capacità nell'ordine dei petabyte.

Esistono numerosi strumenti forniti da partner AWS e terze parti che consentono aggregazione, elaborazione, archiviazione e analisi. Questi strumenti includono New Relic, Splunk, Loggly, Logstash, CloudHealth e Nagios. Tuttavia, la generazione esterna di log di sistema e applicazioni è univoca per ciascun provider di servizi cloud e spesso per ciascun servizio.

Una parte spesso trascurata del processo di monitoraggio è la gestione dei dati. È necessario determinare i requisiti di conservazione per il monitoraggio dei dati, quindi applicare le policy del ciclo di vita di conseguenza. Amazon S3 supporta la gestione del ciclo di vita a livello di bucket S3. Questa gestione del ciclo di vita può essere applicata in modo diverso ai diversi percorsi nel bucket. Verso la fine del ciclo di vita, è possibile trasferire i dati ad Amazon Glacier per lo storage a lungo termine e in seguito la scadenza al termine del periodo di conservazione. La classe di storage S3 Intelligent-Tiering è progettata per ottimizzare i costi trasferendo automaticamente i dati nel livello di accesso più conveniente, senza impatto sulle prestazioni o sovraccarico operativo.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

- Gli Approfondimenti di CloudWatch Logs consentono di eseguire ricerche interattive e analizzare i dati di log in Amazon CloudWatch Logs.
  - Analyzing Log Data with CloudWatch Logs Insights
  - Query di esempio di Approfondimenti di Amazon CloudWatch Logs
- Utilizza Amazon CloudWatch Logs per inviare log ad Amazon S3, dove è possibile utilizzare Amazon Athena per eseguire query sui dati
  - Come posso analizzare i log di accesso al server Amazon S3 tramite Athena?
    - Crea una policy del ciclo di vita di S3 per il bucket dei log di accesso al server. Configura la policy del ciclo di vita per rimuovere periodicamente i file di log. In questo modo, si riduce la quantità di dati analizzati da Athena per ogni query.
      - Come posso creare una policy del ciclo di vita per un bucket S3?

Risorse

Documenti correlati:

- Query di esempio di Approfondimenti di Amazon CloudWatch Logs
- Analyzing Log Data with CloudWatch Logs Insights
- Debugging with Amazon CloudWatch Synthetics and AWS X-Ray
- Come posso creare una policy del ciclo di vita per un bucket S3?
- Come posso usare Amazon Athena per analizzare i log di accesso al server Amazon S3?
- One Observability Workshop
- The Amazon Builders' Library: strumentazione di sistemi distribuiti per visibilità operativa

REL06-BP06 Revisione periodica dell'ambito e delle metriche di monitoraggio

Esegui revisioni frequenti della modalità di implementazione del monitoraggio del carico di lavoro e aggiornarla in base all'evoluzione del carico di lavoro e della relativa architettura. Gli audit regolari del monitoraggio aiutano a ridurre il rischio di indicatori di problemi mancati o trascurati e aiutano ulteriormente il carico di lavoro a raggiungere gli obiettivi di disponibilità.

Un monitoraggio efficace si basa su metriche aziendali chiave, che si evolvono in base al cambiamento delle priorità aziendali. Il processo di revisione del monitoraggio deve porre l'accento sugli indicatori del livello di servizio (SLI) e incorporare le informazioni approfondite provenienti dall'infrastruttura, dalle applicazioni, dai client e dagli utenti.

Risultato desiderato: disponi di una strategia di monitoraggio efficace che viene regolarmente rivista e aggiornata periodicamente, oltre che dopo qualsiasi evento o cambiamento significativo. Verifichi che gli indicatori chiave di integrità dell'applicazione siano ancora pertinenti con l'evoluzione del carico di lavoro e dei requisiti aziendali.

Anti-pattern comuni:

- Raccogli solo metriche predefinite.
- Hai impostato una strategia di monitoraggio, ma non esegui mai alcuna revisione.
- Non discuti il monitoraggio quando vengono distribuite modifiche importanti.
- Fai affidamento a metriche obsolete per determinare l'integrità del carico di lavoro.
- I team operativi sono sommersi da avvisi falsi positivi dovuti a metriche e soglie obsolete.
- Manca l'osservabilità dei componenti dell'applicazione che non vengono monitorati.
- Ti concentri solo su metriche tecniche di basso livello ed escludi le metriche aziendali dal monitoraggio.

Vantaggi dell'adozione di questa best practice: una revisione regolare del monitoraggio consente di anticipare i potenziali problemi e di verificare la capacità di rilevarli. Inoltre, consente di scoprire i punti ciechi che potrebbero essere sfuggiti durante le revisioni precedenti, migliorando ulteriormente la capacità di individuare i problemi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Rivedi le metriche di monitoraggio e l'ambito durante il processo di revisione della prontezza operativa (ORR). Esegui periodicamente revisioni della prontezza operativa per valutare se ci sono lacune tra il carico di lavoro attuale e il monitoraggio configurato. Stabilisci una cadenza regolare per le revisioni delle prestazioni operative e la condivisione delle conoscenze per migliorare la capacità di ottenere prestazioni più elevate dai team operativi. Convalida se le soglie di allarme esistenti sono ancora adeguate e verifica le situazioni in cui i team operativi ricevono avvisi falsi positivi o non monitorano aspetti dell'applicazione che invece devono esserlo.

Il framework di analisi della resilienza fornisce indicazioni utili che possono aiutare a esplorare il processo. L'obiettivo del framework è identificare le potenziali modalità di errore e i controlli preventivi e correttivi da utilizzare per mitigare l'impatto. Queste conoscenze possono aiutare a identificare le metriche e gli eventi giusti da monitorare e segnalare.

Passaggi dell'implementazione

1. Pianifica ed effettua revisioni periodiche dei pannelli di controllo del carico di lavoro. La frequenza può essere diversa a seconda di quanto l'ispezione sia approfondita.
2. Ispeziona l'andamento nei parametri. Confronta i valori dei parametri con i valori storici per vedere se ci sono tendenze che potrebbero suggerire l'esame di un particolare aspetto. Esempi di questo tipo sono l'aumento della latenza, la riduzione della funzione aziendale primaria e l'aumento delle risposte agli errori.
3. Esamina i valori anomali e le anomalie nelle metriche, che possono essere mascherate dalle medie o dalle mediane. Osserva i valori più alti e più bassi durante l'arco temporale e indaga sulle cause di osservazioni che sono molto al di fuori dei limiti normali. Continuando a eliminare queste cause, puoi restringere i limiti delle metriche previste in risposta al miglioramento della coerenza delle prestazioni del carico di lavoro.
4. Ricerca di bruschi cambiamenti nel comportamento. Un cambiamento immediato nella quantità o nella direzione di una metrica può indicare che si è verificato un cambiamento nell'applicazione o nei fattori esterni che potrebbe richiedere l'aggiunta di ulteriori metriche da monitorare.

5. Verifica se l'attuale strategia di monitoraggio rimane pertinente per l'applicazione. Sulla base di un'analisi degli incidenti precedenti (o del framework di analisi della resilienza), valuta se ci sono ulteriori aspetti dell'applicazione che dovrebbero essere incorporati nell'ambito del monitoraggio.
6. Esamina le metriche di monitoraggio dell'utente reale (RUM, Real User Monitoring) per determinare se ci sono lacune nella copertura delle funzionalità dell'applicazione.
7. Rivedi il processo di gestione del cambiamento. Se necessario, aggiorna le procedure per includere una fase di analisi di monitoraggio da eseguire prima di approvare una modifica.
8. Implementa la revisione del monitoraggio come parte della revisione della prontezza operativa e dei processi di correzione degli errori.

Risorse

Best practice correlate:

- REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)
- REL06-BP02 Definizione e calcolo dei parametri (aggregazione)
- REL06-BP07 Monitoraggio del tracciamento end-to-end delle richieste attraverso il sistema
- REL12-BP02 Esecuzione di analisi post-incidente
- REL12-BP06 Esecuzione regolare di GameDay

Documenti correlati:

- Why you should develop a correction of error (COE)
- Using Amazon CloudWatch Dashboards
- Building dashboards for operational visibility
- Advanced Multi-AZ Resilience Patterns - Gray failures
- Query di esempio di Approfondimenti di Amazon CloudWatch Logs
- Debugging with Amazon CloudWatch Synthetics and AWS X-Ray
- One Observability Workshop
- The Amazon Builders' Library: strumentazione di sistemi distribuiti per visibilità operativa
- Using Amazon CloudWatch Dashboards
- AWS Observability Best Practices

- Resilience Analysis Framework
- Resilience Analysis Framework - Observability
- Operational Readiness Review - ORR

REL06-BP07 Monitoraggio del tracciamento end-to-end delle richieste attraverso il sistema

Tieni traccia delle richieste durante l'elaborazione dei componenti del servizio in modo che i team del prodotto possano analizzare i problemi, semplificarne il debug e migliorare le prestazioni.

Risultato desiderato: i carichi di lavoro con tracciabilità completa su tutti i componenti sono caratterizzati da processi di debug più semplici, con conseguente miglioramento del tempo medio di risoluzione (MTTR) degli errori e della latenza, grazie a una più semplice individuazione della causa principale. La tracciabilità end-to-end riduce il tempo necessario per individuare i componenti interessati e approfondire in dettaglio le cause principali degli errori o della latenza.

Anti-pattern comuni:

- Il tracciamento viene utilizzato per alcuni componenti ma non per tutti. Ad esempio, senza il tracciamento AWS Lambda, i team potrebbero non avere una chiara comprensione della latenza causata dagli avviamenti a freddo in un periodo di picco del carico di lavoro.
- I canary Synthetics o le metriche RUM (Real-User Monitoring) non sono configurati con il tracciamento. Senza canary o metriche RUM, la telemetria delle interazioni dei clienti viene omessa dall'analisi dei tracciamenti e ciò rende incompleto il profilo delle prestazioni.
- I carichi di lavoro ibridi includono strumenti di tracciamento cloud-native (nativi del cloud) e di terze parti, ma non sono state prese misure specifiche per selezionare e integrare completamente un'unica soluzione di tracciamento. In base alla soluzione di tracciamento scelta, gli SDK di tracciamento cloud-native (nativi del cloud) devono essere utilizzati per instrumentare i componenti non cloud-native (nativi del cloud) oppure è necessario configurare strumenti di terze parti per acquisire i dati telemetrici delle tracce nativi del cloud.

Vantaggi dell'adozione di questa best practice: quando vengono avvisati della presenza di problemi, i team di sviluppo possono visualizzare un quadro completo delle interazioni tra i componenti del sistema, inclusa la correlazione componente per componente con creazione di log, prestazioni e guasti. Poiché il tracciamento semplifica l'identificazione visiva delle cause principali, viene dedicato meno tempo all'individuazione di tali cause. I team che hanno una visione dettagliata delle interazioni tra i componenti prendono decisioni migliori e più rapide durante la fase di risoluzione dei problemi. Le decisioni, ad esempio quando invocare il failover del disaster recovery (DR) o dove implementare

in modo più efficace le strategie di riparazione automatica, possono essere migliorate analizzando le tracce dei sistemi; ciò ottimizza in ultima analisi la soddisfazione dei clienti nei confronti dei servizi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I team che gestiscono le applicazioni distribuite possono utilizzare strumenti di tracciamento per definire un identificatore di correlazione, raccogliere le tracce delle richieste e creare mappe di servizio dei componenti connessi. Tutti i componenti dell'applicazione devono essere inclusi nelle tracce delle richieste, inclusi client di servizio, gateway middleware (software intermediario) e router di eventi, componenti di elaborazione e archiviazione, tra cui gli archivi e i database dei valori chiave. Includi canary Synthetics o metriche RUM (Real-User Monitoring) nella configurazione del tracciamento end-to-end per misurare le interazioni e la latenza dei client remoti in modo da poter valutare con precisione le prestazioni dei tuoi sistemi rispetto agli accordi sul livello di servizio (SLA) e agli obiettivi corrispondenti.

Puoi utilizzare AWS X-Ray e i servizi di strumentazione del monitoraggio delle applicazioni Amazon CloudWatch per una visione completa delle richieste mentre vengono inviate alla tua applicazione. X-Ray raccoglie la telemetria delle applicazioni e consente di visualizzarla e filtrarla tra payload, funzioni, tracce, servizi, API. La telemetria può essere attivata per componenti di sistema senza codice o a uso limitato di codice. Il monitoraggio delle applicazioni CloudWatch comprende Servicelens per l'integrazione delle tracce con parametri, log e allarmi. La funzionalità Monitoraggio delle applicazioni CloudWatch include anche elementi Synthetics per monitorare gli endpoint e le API, oltre alle metriche RUM (Real-User Monitoring) per dotare di strumenti i client delle applicazioni Web.

Passaggi dell'implementazione

- Utilizza AWS X-Ray su tutti i servizi nativi supportati, come Amazon S3, AWS Lambda e Gateway Amazon API. Questi servizi AWS consentono a X-Ray di attivare opzioni di configurazione utilizzando il modello Infrastructure as code, AWS SDK o la Console di gestione AWS.
- Dota di strumenti le applicazioni AWS Distro for Open Telemetry e X-Ray o gli agenti di raccolta di terze parti.
- Consulta la Guida per gli sviluppatori AWS X-Ray per l'implementazione specifica del linguaggio di programmazione. Queste sezioni della documentazione descrivono come instrumentare le richieste HTTP, le query SQL e altri processi specifici del linguaggio di programmazione delle applicazioni.
- Usa il tracciamento X-Ray per i canary sintetici di Amazon CloudWatch e Amazon CloudWatch RUM al fine di analizzare il percorso delle richieste dal client dell'utente finale lungo l'infrastruttura AWS a valle.

- Configura le metriche e gli allarmi CloudWatch in base allo stato delle risorse e alla telemetria dei canary in modo che i team siano avvisati tempestivamente in merito ai problemi e possano, quindi, analizzare in dettaglio le tracce e le mappe dei servizi con ServiceLens.
- Abilita l'integrazione X-Ray per gli strumenti di tracciamento di terze parti come Datadog, New Relic o Dynatrace in caso di utilizzo di strumenti di terze parti per la tua soluzione di tracciamento principale.

Risorse

Best practice correlate:

- REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

Documenti correlati:

- Cos'è AWS X-Ray?
- Amazon CloudWatch: monitoraggio delle applicazioni
- Debugging with Amazon CloudWatch Synthetics and AWS X-Ray
- The Amazon Builders' Library: strumentazione di sistemi distribuiti per visibilità operativa
- Integrating AWS X-Ray with other AWS services
- AWS Distro for OpenTelemetry and AWS X-Ray
- Amazon CloudWatch: utilizzo del monitoraggio sintetico
- Amazon CloudWatch: utilizzo di CloudWatch RUM
- Set up Amazon CloudWatch synthetics canary and Amazon CloudWatch alarm
- Availability and Beyond: Understanding and Improving the Resilience of Distributed Systems on AWS

Esempi correlati:

- One Observability Workshop

Video correlati:

- AWS re:Invent 2022 - How to monitor applications across multiple accounts

- How to Monitor your AWS Applications

Strumenti correlati:

- AWS X-Ray
- Amazon CloudWatch
- Amazon Route 53

REL 7. Come si progetta il carico di lavoro per adattarsi ai cambiamenti della domanda?

Un carico di lavoro scalabile garantisce l'elasticità per aggiungere o rimuovere risorse in automatico, in modo che sussista una stretta corrispondenza con la domanda attuale in un dato momento.

Best practice

- REL07-BP01 Utilizzo dell'automazione per l'acquisizione o il dimensionamento delle risorse
- REL07-BP02 Ottenimento di risorse quando viene rilevata la compromissione di un carico di lavoro
- REL07-BP03 Ottenimento di risorse dopo aver rilevato che sono necessarie più risorse per un carico di lavoro
- REL07-BP04 Load Testa il tuo carico di lavoro

REL07-BP01 Utilizzo dell'automazione per l'acquisizione o il dimensionamento delle risorse

La definizione, il provisioning e la gestione programmatici dell'infrastruttura e delle risorse sono una pietra miliare dell'affidabilità nel cloud. L'automazione aiuta a semplificare il provisioning delle risorse, facilitare implementazioni coerenti e sicure e scalare le risorse nell'intera infrastruttura.

Risultato desiderato: gestisci infrastructure as code (IaC). Il codice dell'infrastruttura viene definito e gestito nei sistemi di controllo delle versioni (VCS). Deleghi il provisioning delle risorse AWS a meccanismi automatici e sfrutti i servizi gestiti come Application Load Balancer (ALB), Network Load Balancer (NLB) e i gruppi Auto Scaling. Il provisioning delle risorse si avvale di pipeline di integrazione continua/distribuzione continua (CI/CD) in modo che le modifiche al codice avviino automaticamente gli aggiornamenti delle risorse, inclusi gli aggiornamenti delle configurazioni di dimensionamento automatico.

Anti-pattern comuni:

- Distribuisci le risorse manualmente utilizzando la riga di comando o la Console di gestione AWS (conosciuto anche come ClickOps).
- Accoppi strettamente i componenti o le risorse dell'applicazione, creando di conseguenza architetture poco flessibili.
- Implementi policy di dimensionamento rigide che non si adattano ai requisiti aziendali in evoluzione, ai modelli di traffico o ai nuovi tipi di risorse.
- Esegui la stima manuale della capacità di soddisfare la domanda prevista.

Vantaggi della definizione di questa best practice: l'infrastructure as code (IaC) consente di definire l'infrastruttura a livello di programmazione. Questo aiuta a gestire le modifiche all'infrastruttura attraverso lo stesso ciclo di vita dello sviluppo software delle modifiche all'applicazione, favorendo la coerenza e la ripetibilità e riducendo il rischio di attività manuali soggette a errori. Il processo di provisioning e aggiornamento delle risorse può essere semplificato ulteriormente implementando IaC con pipeline di distribuzione automatiche. È possibile implementare gli aggiornamenti dell'infrastruttura in modo affidabile ed efficiente, senza bisogno di interventi manuali. Questa agilità è particolarmente importante quando si tratta di scalare le risorse per soddisfare richieste fluttuanti.

È possibile ottenere una scalabilità dinamica e automatizzata delle risorse in combinazione con IaC e pipeline di distribuzione. Monitorando le metriche chiave e applicando policy di dimensionamento predefinite, il dimensionamento automatico è in grado di effettuare automaticamente il provisioning o il deprovisioning delle risorse secondo necessità, migliorando le prestazioni e l'efficienza in termini di costi. In questo modo si riduce il potenziale di errori manuali o di ritardi in risposta alle modifiche dei requisiti delle applicazioni o del carico di lavoro.

La combinazione di IaC, pipeline di distribuzione automatizzate e dimensionamento automatico consente alle organizzazioni di effettuare il provisioning, l'aggiornamento e il dimensionamento dei propri ambienti in tutta tranquillità. Questa automazione è essenziale per mantenere un'infrastruttura cloud reattiva, resiliente e gestita in modo efficiente.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per configurare l'automazione con le pipeline CI/CD e infrastructure as code (IaC) per l'architettura AWS, scegli un sistema di controllo delle versioni come Git per archiviare i modelli e la configurazione IaC. Questi modelli possono essere scritti utilizzando strumenti quali AWS CloudFormation. Per iniziare, definisci i componenti dell'infrastruttura (ad esempio, AWS VPC, gruppi Amazon EC2 Auto Scaling e database Amazon RDS) all'interno di questi modelli.

Successivamente, integra questi modelli IaC con una pipeline CI/CD per automatizzare il processo di implementazione. AWS CodePipeline fornisce una soluzione AWS nativa senza soluzione di continuità oppure puoi utilizzare altre soluzioni CI/CD di terze parti. Crea una pipeline che si attivi quando vengono apportate modifiche al repository di controllo delle versioni. Configura la pipeline in modo da includere le fasi di lint e validazione dei modelli IaC, distribuisci l'infrastruttura in un ambiente di gestione temporanea, esegui i test automatici e infine distribuisci in produzione. Incorpora le fasi di approvazione ove necessario per mantenere il controllo sulle modifiche. Questa pipeline automatica non solo accelera l'implementazione, ma facilita anche la coerenza e l'affidabilità tra gli ambienti.

Configura il dimensionamento automatico di risorse come istanze Amazon EC2, attività Amazon ECS e repliche di database nell'ambiente IaC per garantire aumento e riduzione orizzontale secondo necessità. Questo approccio migliora la disponibilità e le prestazioni delle applicazioni e ottimizza i costi regolando dinamicamente le risorse in base alla domanda. Per un elenco delle risorse supportate, consulta Amazon EC2 Auto Scaling e AWS Auto Scaling.

Passaggi dell'implementazione

1. Crea e utilizza un repository del codice sorgente per archiviare il codice che controlla la configurazione dell'infrastruttura. Esegui il commit delle modifiche a questo repository per riflettere eventuali modifiche in corso da apportare.
2. Seleziona una soluzione infrastructure as code (IaC) come AWS CloudFormation per mantenere l'infrastruttura aggiornata e rilevare le incoerenze (deviazione) rispetto allo stato previsto.
3. Integra la piattaforma IaC con la pipeline CI/CD per automatizzare le implementazioni.
4. Determina e raccogli le metriche appropriate per il dimensionamento automatico delle risorse.
5. Configura il dimensionamento automatico delle risorse utilizzando policy appropriate per aumentare orizzontalmente e ridurre orizzontalmente le risorse per i componenti del carico di lavoro. Considera l'utilizzo di un dimensionamento pianificato per modelli di utilizzo prevedibili.
6. Monitora le implementazioni per rilevare guasti e regressioni. Implementa meccanismi di ripristino dello stato precedente all'interno della piattaforma CI/CD per annullare le modifiche, se necessario.

Risorse

Documenti correlati:

- AWS Auto Scaling: How Scaling Plans Work
- Marketplace AWS: prodotti che possono essere utilizzati con Auto Scaling

- Managing Throughput Capacity Automatically with DynamoDB Auto Scaling
- Uso di un bilanciatore del carico con un gruppo Auto Scaling
- What Is AWS Global Accelerator?
- What Is Amazon EC2 Auto Scaling?
- Cos'è AWS Auto Scaling?
- What is Amazon CloudFront?
- What is Amazon Route 53?
- Cos'è l'Elastic Load Balancing?
- Cos'è un Network Load Balancer?
- Cos'è un Application Load Balancer?
- Integrating Jenkins with AWS CodeBuild and AWS CodeDeploy
- Creating a four stage pipeline with AWS CodePipeline

Video correlati:

- Back to Basics: Deploy Your Code to Amazon EC2
- AWS Supports You | Starting Your Infrastructure as Code Solution Using AWS CloudFormation Templates
- Streamline Your Software Release Process Using AWS CodePipeline
- Monitor AWS Resources Using Amazon CloudWatch Dashboards
- Create Cross Account & Cross Region CloudWatch Dashboards | Amazon Web Services

REL07-BP02 Ottenimento di risorse quando viene rilevata la compromissione di un carico di lavoro

All'occorrenza, procedi a scalare le risorse in modo reattivo se la disponibilità è influenzata per ripristinare la disponibilità del carico di lavoro.

Devi prima configurare il controllo dell'integrità e i criteri su questi controlli per indicare quando la disponibilità è influenzata dalla mancanza di risorse. Quindi invita il personale appropriato a scalare manualmente la risorsa o attivare l'automazione per dimensionarla automaticamente.

Il dimensionamento può essere regolato manualmente in base al carico di lavoro, ad esempio modificando il numero di istanze EC2 in un gruppo Auto Scaling o modificando il throughput di una tabella DynamoDB tramite la Console di gestione AWS o AWS CLI). Tuttavia, è opportuno ricorrere

all'automazione ogni volta che è possibile (consulta Utilizzo dell'automazione per l'acquisizione o il dimensionamento delle risorse).

Risultato desiderato: avvio di operazioni di dimensionamento (in automatico o manualmente) per il ripristino della disponibilità in caso di rilevamento di un guasto o di un peggioramento dell'esperienza del cliente.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Implementa l'osservabilità e il monitoraggio su tutti i componenti del carico di lavoro, per monitorare l'esperienza del cliente e rilevare i guasti. Definisci le procedure, manuali o automatizzate, per scalare le risorse richieste. Per ulteriori informazioni, consulta REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti.

Passaggi dell'implementazione

- Definisci le procedure (manuali o automatiche) per scalare le risorse richieste.
  - Le procedure di dimensionamento dipendono da come sono progettati i diversi componenti del carico di lavoro.
  - Le procedure di dimensionamento variano anche a seconda della tecnologia sottostante utilizzata.
    - I componenti che utilizzano AWS Auto Scaling possono impiegare piani di dimensionamento per configurare una serie di istruzioni per scalare le risorse. Se si lavora con AWS CloudFormation o si aggiungono tag a risorse AWS, è possibile impostare piani di dimensionamento per diversi set di risorse, per applicazione. Auto Scaling fornisce raccomandazioni per strategie di dimensionamento personalizzate per ogni risorsa. Dopo aver creato il piano di dimensionamento, Auto Scaling combina i metodi di dimensionamento dinamico e predittivo per supportare la tua strategia di dimensionamento. Per ulteriori informazioni, consulta How scaling plans work.
    - Amazon EC2 Auto Scaling verifica la disponibilità del numero corretto di istanze Amazon EC2 per gestire il carico dell'applicazione. È possibile creare raccolte di istanze EC2, denominate gruppi Auto Scaling. Puoi specificare il numero minimo e massimo di istanze in ogni gruppo Auto Scaling. Amazon EC2 Auto Scaling garantisce che il gruppo non superi mai o scenda al di sotto di questi limiti. Per ulteriori informazioni, consulta What is Amazon EC2 Auto Scaling?
    - La scalabilità automatica di Amazon DynamoDB utilizza il servizio Application Auto Scaling per regolare in modo dinamico la capacità effettiva di trasmissione allocata per conto tuo in

risposta ai modelli di traffico effettivi. In tal modo una tabella o un indice secondario globale può aumentare la capacità di lettura e scrittura allocata per gestire improvvisi aumenti di traffico, senza limitazione (della larghezza di banda della rete). Per ulteriori dettagli, consulta Managing throughput capacity automatically with DynamoDB auto scaling.

Risorse

Best practice correlate:

- REL07-BP01 Utilizzo dell'automazione per l'acquisizione o il dimensionamento delle risorse
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

Documenti correlati:

- AWS Auto Scaling: How Scaling Plans Work
- Managing Throughput Capacity Automatically with DynamoDB Auto Scaling
- What Is Amazon EC2 Auto Scaling?

REL07-BP03 Ottenimento di risorse dopo aver rilevato che sono necessarie più risorse per un carico di lavoro

Una delle caratteristiche più preziose del cloud computing è la capacità di fornire risorse in modo dinamico.

Negli ambienti di calcolo tradizionali on-premises, è necessario identificare e fornire in anticipo la capacità sufficiente per soddisfare i picchi di domanda. Questo è un problema perché è costoso e perché comporta rischi per la disponibilità se si sottovalutano le esigenze di capacità di picco del carico di lavoro.

Nel cloud non è necessario farlo. Invece, è possibile allocare capacità di calcolo, database e altre risorse in base alle esigenze per soddisfare la domanda attuale e prevista. Soluzioni automatizzate come Amazon EC2 Auto Scaling e Application Auto Scaling possono portare automaticamente le risorse online in base a metriche specificate dall'utente. Ciò può rendere il processo di dimensionamento più semplice e prevedibile e può rendere il carico di lavoro significativamente più affidabile garantendo che siano sempre disponibili risorse sufficienti.

Risultato desiderato: configuri il dimensionamento automatico delle risorse di calcolo e di altro tipo per soddisfare la domanda. Nelle policy di dimensionamento fornisci un margine sufficiente per consentire la gestione di picchi di traffico mentre vengono messe online altre risorse.

Anti-pattern comuni:

- Fornisci un numero fisso di risorse scalabili.
- Scegli una metrica di dimensionamento che non è correlata alla domanda effettiva.
- I piani di dimensionamento non prevedono un margine sufficiente per supportare picchi di domanda.
- Le policy di dimensionamento aggiungono capacità troppo tardi, con conseguente esaurimento della capacità e degrado del servizio mentre altre risorse vengono messe online.
- Non riesci a configurare correttamente il conteggio delle risorse minime e massime, con conseguenti errori di dimensionamento.

Vantaggi dell'adozione di questa best practice: disporre di risorse sufficienti per soddisfare la domanda attuale è fondamentale per fornire un'elevata disponibilità del carico di lavoro e rispettare gli obiettivi di livello di servizio (SLO) definiti. Il dimensionamento automatico consente di fornire la giusta quantità di calcolo, database e altre risorse necessarie al carico di lavoro per soddisfare la domanda attuale e prevista. Non è necessario determinare le esigenze di capacità di picco e allocare staticamente le risorse per soddisfarle. Invece, man mano che la domanda cresce, è possibile allocare più risorse per soddisfarla e, quando la domanda diminuisce, è possibile disattivare le risorse per ridurre i costi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

In primo luogo, è necessario determinare se il componente del carico di lavoro è adatto al dimensionamento automatico. Questi componenti sono chiamati a scalabilità orizzontale perché forniscono le stesse risorse e si comportano in modo identico. Esempi di componenti a scalabilità orizzontale includono le istanze EC2 configurate allo stesso modo, attività Amazon Elastic Container Service (ECS) e pod in esecuzione su Amazon Elastic Kubernetes Service (EKS). Queste risorse di calcolo sono in genere collocate dietro un bilanciatore del carico e vengono chiamate repliche.

Altre risorse replicate possono includere repliche di lettura del database, tabelle Amazon DynamoDB e cluster Amazon ElastiCache (Redis OSS). Per un elenco completo delle risorse supportate, consulta la pagina relativa ai servizi AWS che puoi utilizzare con Application Auto Scaling.

Per le architetture basate su container, potrebbe essere necessario scalare in due modi diversi. In primo luogo, potrebbe essere necessario scalare i container che forniscono servizi scalabili orizzontalmente. In secondo luogo, potrebbe essere necessario scalare le risorse di calcolo per creare spazio per nuovi container. Per ogni livello esistono diversi meccanismi di dimensionamento automatico. Per scalare le attività ECS, puoi utilizzare Application Auto Scaling. Per scalare i pod Kubernetes, puoi utilizzare Horizontal Pod Autoscaler (HPA) o Kubernetes Event-driven Autoscaling (KEDA). Per scalare le risorse di calcolo, puoi utilizzare Provider di capacità per ECS o puoi utilizzare Karpenter o Cluster Autoscaler per Kubernetes.

Quindi, seleziona la modalità di esecuzione del dimensionamento automatico. Esistono tre opzioni principali: dimensionamento basato su metriche, dimensionamento pianificato e dimensionamento predittivo.

Dimensionamento basato su metriche

Il dimensionamento basato su metriche fornisce le risorse in base al valore di una o più metriche di dimensionamento. Una metrica di dimensionamento è quella che corrisponde alla domanda del carico di lavoro. Un buon modo per determinare le metriche di dimensionamento appropriate è quello di eseguire test di carico in un ambiente non di produzione. Durante i test di carico, mantieni fisso il numero di risorse scalabili e aumenta lentamente la domanda (ad esempio, il throughput, la simultaneità o gli utenti simulati). Cerca quindi le metriche che aumentano (o diminuiscono) con l'aumento della domanda e, viceversa, che diminuiscono (o aumentano) con il calo della domanda. Le metriche di dimensionamento tipiche includono l'utilizzo della CPU, la profondità della coda di lavoro (ad esempio una coda Amazon SQS), il numero di utenti attivi e il throughput di rete.

> **Note**
>
> AWS ha osservato che con la maggior parte delle applicazioni, l'utilizzo della memoria aumenta durante la preparazione dell'applicazione per poi raggiunge un valore costante. Quando la domanda diminuisce, l'utilizzo della memoria rimane in genere elevato anziché diminuire in parallelo. Poiché l'utilizzo della memoria non corrisponde alla domanda in entrambe le direzioni, ovvero cresce e diminuisce con la domanda, valuta attentamente questa metrica prima di selezionarla per il dimensionamento automatico.

Il dimensionamento basato sulle metriche è un'operazione latente. Le metriche di utilizzo possono impiegare diversi minuti per propagarsi ai meccanismi di dimensionamento automatico, che in genere attendono un chiaro segnale di aumento della domanda prima di reagire. Man mano che l'autoscaler

crea nuove risorse, può essere necessario del tempo aggiuntivo per raggiungere il pieno servizio. Per questo motivo, è importante non impostare i target delle metriche di dimensionamento troppo vicini all'utilizzo completo (ad esempio, 90% di utilizzo della CPU). Così facendo, si rischia di esaurire la capacità delle risorse esistenti prima che una capacità aggiuntiva possa essere messa online. Gli obiettivi tipici di utilizzo delle risorse possono variare tra il 50 e il 70% per una disponibilità ottimale, a seconda dei modelli di domanda e del tempo necessario per effettuare il provisioning di risorse aggiuntive.

Dimensionamento pianificato

Il dimensionamento pianificato fornisce o rimuove le risorse in base al calendario o all'ora del giorno. È spesso utilizzato per i carichi di lavoro che hanno una domanda prevedibile, come i picchi di utilizzo durante le ore di lavoro nei giorni feriali o gli eventi di vendita. Sia Amazon EC2 Auto Scaling che Application Auto Scaling supportano il dimensionamento pianificato. Il cron scaler di KEDA supporta il dimensionamento pianificato dei pod Kubernetes.

Dimensionamento predittivo

Il dimensionamento predittivo utilizza il machine learning per scalare automaticamente le risorse in base alla domanda prevista. Il dimensionamento predittivo analizza il valore storico di una metrica di utilizzo fornita dall'utente e ne prevede continuamente il valore futuro. Il valore previsto viene quindi utilizzato per scalare la risorsa verso l'alto o verso il basso. Amazon EC2 Auto Scaling può eseguire il dimensionamento predittivo.

Passaggi dell’implementazione

1. Determina se il componente del carico di lavoro è adatto al dimensionamento automatico.
2. Determina il tipo di meccanismo di dimensionamento più appropriato per il carico di lavoro: dimensionamento basato sulle metriche, dimensionamento pianificato o dimensionamento predittivo.
3. Seleziona il meccanismo di dimensionamento automatico appropriato per il componente. Per le istanze Amazon EC2, utilizza Amazon EC2 Auto Scaling. Per altri servizi AWS, utilizza Application Auto Scaling. Per i pod Kubernetes (come quelli in esecuzione in un cluster Amazon EKS), prendi in considerazione Horizontal Pod Autoscaler (HPA) o Kubernetes Event-driven Autoscaling (KEDA). Per i nodi Kubernetes o EKS, prendi in considerazione Karpenter e Cluster Auto Scaler (CAS).
4. Per il dimensionamento basato sulle metriche o pianificato, esegui test di carico per determinare le metriche di dimensionamento e i valori di destinazione appropriati per il carico di lavoro. Per il dimensionamento pianificato, determina il numero di risorse necessarie alle date e agli orari

selezionati. Determina il numero massimo di risorse necessarie per servire i picchi di traffico previsti.

5. Configura l'autoscaler in base alle informazioni raccolte in precedenza. Per maggiori dettagli, consulta la documentazione del servizio di dimensionamento automatico. Verifica che i limiti di dimensionamento massimo e minimo siano configurati correttamente.
6. Verifica che la configurazione di dimensionamento funzioni come previsto. Esegui i test di carico in un ambiente non di produzione e osserva come reagisce il sistema, regolandolo se necessario. Quando abiliti il dimensionamento automatico in produzione, configura gli allarmi appropriati per segnalare qualsiasi comportamento imprevisto.

Risorse

Documenti correlati:

- What Is Amazon EC2 Auto Scaling?
- AWS Prescriptive Guidance: Load testing applications
- Marketplace AWS: prodotti che possono essere utilizzati con Auto Scaling
- Managing Throughput Capacity Automatically with DynamoDB Auto Scaling
- Predictive Scaling for EC2, Powered by Machine Learning
- Scheduled Scaling for Amazon EC2 Auto Scaling
- Telling Stories About Little's Law

REL07-BP04 Load Testa il tuo carico di lavoro

Adotta un metodo di test del carico per misurare se l'attività di dimensionamento soddisfa i requisiti del carico di lavoro.

È importante eseguire test di carico prolungati. I test di carico dovrebbero scoprire il punto di rottura e testare le prestazioni del carico di lavoro. AWS semplifica la configurazione di ambienti di test temporanei che modellano la scala del carico di lavoro di produzione. Nel cloud, puoi creare un ambiente di test su scala produttiva on demand, completare i test e disattivare le risorse. Poiché paghi per l'ambiente di test solo quando è in esecuzione, puoi simulare un ambiente live a un costo notevolmente inferiore rispetti ai test on-premises.

I test di carico in produzione dovrebbero anche essere considerati come parte delle giornate di gioco in cui il sistema di produzione viene messo alla prova, durante le ore di utilizzo inferiore del cliente,

con tutto il personale a disposizione per interpretare i risultati e risolvere eventuali problemi che si presentano.

Anti-pattern comuni:

- Eseguire test di carico su implementazioni che non presentano la stessa configurazione della tua produzione.
- Eseguire test di carico solo su singole parti del carico di lavoro e non sulla sua interezza.
- Eseguire test di carico con un sottoinsieme di richieste e non con un set rappresentativo delle richieste effettive.
- Eseguire test di carico su un fattore di sicurezza di poco superiore al carico previsto.

Vantaggi dell'adozione di questa best practice: saprai quali sono i componenti dell'architettura che non funzionano sotto carico e potrai identificare per tempo i parametri che indicano l'avvicinamento al carico in questione, così da affrontare il problema e prevenire l'impatto dell'esito negativo.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

- Esegui test di carico per identificare quali aspetti del carico di lavoro indicano la necessità di aggiungere o rimuovere capacità. Il test di carico deve avere un traffico rappresentativo simile a quello che ricevi nella produzione. Aumenta il carico mentre osservi i parametri implementati per stabilire quale di questi indica quando è necessario aggiungere o rimuovere risorse.
  - Test di carico distribuito su AWS: simula migliaia di utenti connessi
    - Identifica la combinazione di richieste. Potresti avere diverse combinazioni di richieste, quindi dovresti esaminare vari intervalli di tempo per identificare la combinazione di traffico.
    - Implementa un driver di caricamento. Puoi utilizzare codice personalizzato, software open source o software commerciale per implementare un driver di carico.
    - Esegui un test di carico iniziale con una capacità ridotta. Puoi vedere alcuni effetti immediati applicando il carico su una capacità inferiore, possibilmente pari a un'istanza o a un container.
    - Esegui un test di carico con una capacità maggiore. Gli effetti saranno diversi su un carico distribuito, quindi è necessario eseguire il test in condizioni quanto più simili possibili all'ambiente del prodotto.

Risorse

Documenti correlati:

- Test di carico distribuito su AWS: simula migliaia di utenti connessi
- Load testing applications

Video correlati:

- AWS Summit ANZ 2023: accelera con sicurezza grazie ai test di carico AWS distribuiti

REL 8. In che modo implementare le modifiche?

Per implementare nuove funzionalità e verificare che i carichi di lavoro e l'ambiente operativo eseguano software noti e che sia possibile applicare patch o sostituirli in modo prevedibile, sono necessarie modifiche controllate. Se invece non sono controllate, risulta difficile prevederne l'effetto o risolvere eventuali problemi che causano.

Best practice

- REL08-BP01 Utilizzo di runbook per attività standard come l'implementazione
- REL08-BP02 Esecuzione di test funzionali come parte integrante dell'implementazione
- REL08-BP03 Esecuzione di test di resilienza come parte integrante dell'implementazione
- REL08-BP04 Esecuzione dell'implementazione utilizzando un'infrastruttura immutabile
- REL08-BP05 Implementazione delle modifiche tramite automazione

REL08-BP01 Utilizzo di runbook per attività standard come l'implementazione

I runbook sono le procedure predefinite per ottenere risultati specifici. Utilizza i runbook per eseguire attività standard, o manualmente o automaticamente. Alcuni esempi includono l'implementazione di un carico di lavoro, l'applicazione di patch a un carico di lavoro o la realizzazione di modifiche DNS.

Ad esempio, mettere in atto processi per garantire la sicurezza del rollback durante le implementazioni. Garantire la possibilità di eseguire il rollback di un'implementazione senza interruzioni per i clienti è fondamentale per rendere un servizio affidabile.

Per le procedure di runbook, inizia da un processo manuale valido ed efficace, implementalo nel codice e richiamalo per l'esecuzione automatica, se necessario.

Anche per carichi di lavoro sofisticati e altamente automatizzati, i runbook sono ancora utili per l'esecuzione di giornate di gioco o per soddisfare rigorosi requisiti di reportistica e audit.

Tieni presente che i playbook vengono utilizzati in risposta a incidenti specifici e i runbook vengono utilizzati per ottenere risultati specifici. Spesso, i runbook sono per attività di routine, mentre i playbook vengono utilizzati per rispondere a eventi non di routine.

Anti-pattern comuni:

- Eseguire modifiche impreviste alla configurazione nella produzione.
- Ignorare le fasi del piano per velocizzare l'implementazione, compromettendone la riuscita.
- Apportare modifiche senza testarne l'annullamento.

Vantaggi dell'adozione di questa best practice: la pianificazione efficace aumenta la capacità di eseguire correttamente le modifiche, perché sei a conoscenza di tutti i sistemi interessati. Convalidare le modifiche negli ambienti di test aumenta la tua sicurezza.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

- Fornisci risposte coerenti e tempestive a eventi noti documentando le procedure nei runbook.
- Usa il principio di Infrastructure as code per definire l'infrastruttura Utilizzando AWS CloudFormation o una terza parte affidabile per definire la tua infrastruttura, puoi utilizzare un software per il controllo delle versioni per gestire le versioni e tenere traccia delle modifiche.
  - Utilizza AWS CloudFormation o un provider di terze parti affidabile per definire l'infrastruttura.
    - Cos’è AWS CloudFormation?
  - Crea modelli unici e disaccoppiati, utilizzando solidi principi di progettazione del software.
    - Stabilisci le autorizzazioni, i modelli e le parti responsabili dell'implementazione
      - Controlling access with AWS Identity and Access Management
    - Utilizza un sistema di gestione del codice sorgente ospitato basato su una tecnologia popolare come Git per archiviare il codice sorgente e la configurazione infrastructure as code (IaC).

Risorse

Documenti correlati:

- Partner APN: partner per la creazione di soluzioni di implementazione automatizzate
- Marketplace AWS: prodotti per l'automazione delle implementazioni
- Cos'è AWS CloudFormation?

Esempi correlati:

- Automazione delle operazioni con playbook e runbook

REL08-BP02 Esecuzione di test funzionali come parte integrante dell'implementazione

Utilizza tecniche come i test di unità e i test di integrazione per convalidare le funzionalità richieste.

Il test di unità è un processo in cui si testa la più piccola unità funzionale di codice per convalidarne il comportamento. I test di integrazione cercano di convalidare che ogni funzionalità dell'applicazione operi secondo i requisiti del software. Mentre i test di unità si concentrano sulla verifica di una parte dell'applicazione in modo isolato, i test di integrazione considerano gli effetti collaterali (ad esempio, l'effetto della modifica dei dati attraverso un'operazione di mutazione). In entrambi i casi, i test devono essere integrati in una pipeline di implementazione e, se i criteri di esito positivo non sono soddisfatti, la pipeline viene interrotta o ripristinata. Questi test vengono eseguiti in un ambiente di pre-produzione, gestito per fasi prima della produzione nella pipeline.

Puoi ottenere i migliori risultati quando questi test vengono eseguiti automaticamente come parte delle operazioni di sviluppo e implementazione. Ad esempio, con AWS CodePipeline, gli sviluppatori affidano le modifiche a un repository di origine in cui CodePipeline rileva automaticamente le modifiche. L'applicazione viene creata e i test di unità vengono eseguiti. Dopo che i test di unità sono stati superati, il codice creato viene distribuito sui server di gestione temporanea per il test. Dal server temporaneo, CodePipeline esegue più test, ad esempio test di integrazione o caricamento. Una volta completati con successo i test, CodePipeline distribuisce il codice testato e approvato alle istanze di produzione.

Risultato desiderato: utilizzi l'automazione per eseguire test di unità e di integrazione per verificare che il codice si comporti come previsto. Questi test sono integrati nel processo di implementazione e un errore del test interrompe l'implementazione.

Anti-pattern comuni:

- Ignori o aggiri gli errori di test e i piani durante il processo di implementazione per accelerare la tempistica di implementazione.

- I test vengono eseguiti manualmente al di fuori della pipeline di implementazione.
- Non esegui le fasi di test nell'automazione tramite i flussi di lavoro manuali di emergenza.
- Esegui i test automatici in un ambiente che non assomiglia molto all'ambiente di produzione.
- Crei una suite di test non sufficientemente flessibile e difficile da mantenere, aggiornare o scalare con l'evoluzione dell'applicazione.

Vantaggi dell'adozione di questa best practice: i test automatici durante il processo di implementazione individuano tempestivamente i problemi, riducendo il rischio di un rilascio in produzione con bug o comportamenti imprevisti. I test di unità verificano che il codice si comporti come desiderato e che i contratti API siano rispettati. I test di integrazione convalidano il funzionamento del sistema in base ai requisiti specificati. Questi tipi di test verificano il funzionamento previsto di componenti quali interfacce utente, API, database e codice sorgente.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Adotta un approccio alla scrittura del software basato sullo sviluppo guidato dai test (TDD, Test-Driven Development), in cui sviluppi casi di test per specificare e convalidare il codice. Per iniziare, crea casi di test per ogni funzione. Se il test non va a buon fine, scrivi un nuovo codice per superare il test. Questo approccio consente di convalidare il risultato atteso di ciascuna funzione. Esegui i test di unità e verifica che vengano superati prima di eseguire il commit del codice in un repository del codice sorgente.

Implementa test di unità e di integrazione come parte delle fasi di compilazione, test e implementazione della pipeline CI/CD. Automatizza i test e avvia automaticamente i test ogni volta che una nuova versione dell'applicazione è pronta per essere implementata. Se non si soddisfano i criteri di esito positivo, la pipeline si arresta o viene sottoposta a rollback.

Se l'applicazione è un'app web o per dispositivi mobili, esegui test di integrazione automatizzati su più browser desktop o dispositivi reali. Questo approccio è particolarmente utile per convalidare la compatibilità e la funzionalità delle app per dispositivi mobili su una vasta gamma di dispositivi.

Passaggi dell’implementazione

1. Scrivi test di unità prima di scrivere codice funzionale (sviluppo basato su test o TDD). Stabilisci linee guida per il codice in modo che la scrittura e l'esecuzione di test di unità siano un requisito di codifica non funzionale.

2. Crea una suite di test di integrazione automatizzati che coprano le funzionalità testabili identificate. Questi test devono simulare le interazioni degli utenti e convalidare i risultati attesi.
3. Crea l'ambiente di test necessario per eseguire i test di integrazione. Questo può includere ambienti di gestione temporanea o di pre-produzione che simulano fedelmente l'ambiente di produzione.
4. Configura le fasi di origine, compilazione, test e distribuzione utilizzando la console AWS CodePipeline o AWS Command Line Interface (CLI).
5. Distribuisci l'applicazione una volta che il codice è stato compilato e testato. AWS CodeDeploy può distribuirla negli ambienti di gestione temporanea (test) e di produzione. Questi ambienti possono includere istanze Amazon EC2, funzioni AWS Lambda o server on-premises. Per distribuire l'applicazione in tutti gli ambienti si deve utilizzare lo stesso meccanismo di implementazione.
6. Monitora l'andamento della pipeline e lo stato di ogni fase. Utilizza i controlli di qualità per bloccare la pipeline in base allo stato dei test. Puoi inoltre ricevere notifiche per qualsiasi errore che si verifica durante l'esecuzione o il completamento della pipeline.
7. Monitora costantemente i risultati dei test e cerca modelli, regressioni o aree che richiedono maggiore attenzione. Utilizza queste informazioni per migliorare la suite di test, identificare le aree dell'applicazione che richiedono test più approfonditi e ottimizza il processo di implementazione.

Risorse

Best practice correlate:

- REL07-BP04 Esecuzione di un test di carico sul carico di lavoro
- REL08-BP03 Esecuzione di test di resilienza come parte integrante dell'implementazione
- REL12-BP04 Test della resilienza tramite l'utilizzo dell'ingegneria del caos

Documenti correlati:

- AWS Prescriptive Guidance: Test automation
- Continuous Delivery and Continuous Integration
- Indicators for functional testing
- Monitoring pipelines
- Use AWS CodePipeline with AWS CodeBuild to test code and run builds
- AWS Device Farm

REL08-BP03 Esecuzione di test di resilienza come parte integrante dell'implementazione

Integra i test di resilienza introducendo consapevolmente errori nel sistema per misurarne la capacità in caso di scenari destabilizzanti. I test di resilienza, diversamente dai test funzionali e dagli unit test che di solito sono integrati nei cicli di implementazione, si concentrano sull'identificazione di errori imprevisti nel sistema. Puoi iniziare l'integrazione dei test di resilienza nella fase di pre-produzione, ma stabilisci l'obiettivo di implementare questi test in produzione durante le giornate di gioco.

Risultato desiderato: maggiore fiducia nella capacità del sistema di resistere al degrado nella produzione grazie ai test di resilienza. Gli esperimenti identificano i punti di debolezza che potrebbero causare errori, consentendoti di migliorare il sistema per mitigare automaticamente ed efficacemente errori e danneggiamento.

Anti-pattern comuni:

- Mancanza di osservabilità e monitoraggio nei processi di implementazione.
- Dipendenza dagli esseri umani per risolvere gli errori del sistema.
- Meccanismi di analisi di scarsa qualità.
- Supporto per i problemi noti del sistema e mancanza di sperimentazione per identificare eventuali incognite.
- Identificazione degli errori, ma nessuna risoluzione.
- Nessuna documentazione degli esiti e dei runbook.

Vantaggi dell'adozione delle best practice: i test di resilienza integrati nelle implementazioni consentono di identificare problemi sconosciuti nel sistema che altrimenti passerebbero inosservati, con conseguenti tempi di inattività nella produzione. L'identificazione di queste incognite nel sistema ti consente di documentare gli esiti, integrare i test nel processo CI/CD e creare runbook che semplificano la mitigazione attraverso meccanismi efficienti e ripetibili.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

I moduli di test di resilienza più comuni che possono essere integrati nelle implementazioni del sistema sono il disaster recovery e l'ingegneria del caos.

- Includi gli aggiornamenti ai piani di disaster recovery e alle procedure operative standard (SOP) con qualsiasi implementazione significativa.

- Integra i test di affidabilità nelle pipeline di implementazione automatizzate. Servizi come AWS Resilience Hub possono essere integrati nella pipeline CI/CD al fine di valutare in modo continuo e automatico la resilienza nell'ambito di ogni implementazione.
- Definisci le applicazioni in AWS Resilience Hub. Le valutazioni della resilienza generano frammenti di codice che consentono di creare procedure di ripristino come i documenti di AWS Systems Manager per le applicazioni e forniscono un elenco di controlli e allarmi Amazon CloudWatch consigliati.
- Una volta aggiornati i piani di disaster recovery e le SOP, completa i test di disaster recovery per verificarne l'efficacia. I test di disaster recovery consentono di determinare se è possibile ripristinare il sistema dopo un evento e tornare alle normali operazioni. Puoi simulare varie strategie di disaster recovery e determinare se la pianificazione è sufficiente a soddisfare i requisiti di tempo di attività. Le strategie di disaster recovery più comuni includono backup e ripristino, pilot light, cold standby, warm standby, standby a caldo e attivo-attivo e si differenziano tutte per costi e complessità. Prima dei test di disaster recovery, consigliamo di definire l'obiettivo del tempo di ripristino (RTO) e l'obiettivo del punto di ripristino (RPO) in modo da semplificare la scelta della strategia da simulare. AWS offre strumenti di disaster recovery come AWS Elastic Disaster Recovery, per muovere i primi passi nella pianificazione e nei test.
- Gli esperimenti di ingegneria del caos introducono interruzioni nel sistema, come interruzioni di rete ed errori del servizio. Simulando con gli errori controllati, puoi scoprire le vulnerabilità del sistema contenendo al contempo l'impatto degli errori inseriti. Analogamente alle altre strategie, esegui simulazioni controllate di guasti in ambienti non di produzione, con servizi come AWS Fault Injection Service, per acquisire sicurezza prima dell'implementazione in produzione.

Risorse

Documenti correlati:

- Experiment with failure using resilience testing to build recovery preparedness
- Continually assessing application resilience with AWS Resilience Hub and AWS CodePipeline
- Disaster recovery (DR) architecture on AWS, part 1: Strategies for recovery in the cloud
- Verify the resilience of your workloads using Chaos Engineering
- Principles of Chaos Engineering
- Workshop su Chaos Engineering

Video correlati:

- AWS re:Invent 2020: Testing Resilience using Chaos Engineering
- Improve Application Resilience with AWS Fault Injection Service
- Prepare & Protect Your Applications From Disruption With AWS Resilience Hub

REL08-BP04 Esecuzione dell'implementazione utilizzando un'infrastruttura immutabile

L'infrastruttura immutabile è un modello che richiede che non vengano applicati aggiornamenti, patch di sicurezza o modifiche di configurazione sui carichi di lavoro di produzione. Quando è necessaria una modifica, l'architettura viene costruita su una nuova infrastruttura e distribuita alla produzione.

Segui una strategia di implementazione dell'infrastruttura immutabile per aumentare l'affidabilità, la coerenza e la riproducibilità nelle implementazioni dei carichi di lavoro.

Risultato desiderato: con un'infrastruttura immutabile, non sono consentite modifiche locali (in-place) per l'esecuzione delle risorse dell'infrastruttura all'interno di un carico di lavoro. Invece, quando è necessaria una modifica, un nuovo set di risorse infrastrutturali aggiornate contenente tutte le modifiche necessarie viene implementato in parallelo alle risorse esistenti. Questa implementazione viene convalidata automaticamente e, in caso di successo, il traffico viene gradualmente trasferito al nuovo set di risorse.

Questa strategia di implementazione si applica, ad esempio, agli aggiornamenti software, alle patch di sicurezza, alle modifiche apportate all'infrastruttura, agli aggiornamenti della configurazione e agli aggiornamenti delle applicazioni.

Anti-pattern comuni:

- Implementazione locale (in-place) di modifiche alle risorse dell'infrastruttura in esecuzione.

Vantaggi dell'adozione di questa best practice:

- Maggiore coerenza tra gli ambienti: l'assenza di differenze nelle risorse dell'infrastruttura tra ambienti garantisce l'aumento della coerenza e la semplificazione dei test.
- Riduzione delle deviazioni di configurazione: sostituendo le risorse dell'infrastruttura con una configurazione nota e controllata dalla versione, l'infrastruttura viene impostata a uno stato noto, testato e affidabile, evitando deviazioni di configurazione.
- Implementazioni atomiche affidabili: il completamento delle implementazioni avviene con successo o non cambia nulla, così da aumentare coerenza e affidabilità del processo di implementazione.

- Implementazioni semplificate: le implementazioni sono semplificate poiché non devono supportare gli aggiornamenti. Gli aggiornamenti sono solo nuove implementazioni.
- Implementazioni più sicure con processi di rollback e ripristino rapidi: le implementazioni sono più sicure poiché la versione funzionante precedente non viene modificata. Puoi eseguire il rollback se vengono rilevati errori.
- Potenziamento del profilo di sicurezza: non consentire modifiche all'infrastruttura si traduce nella possibilità di disabilitare i meccanismi di accesso remoto (come SSH). Questo riduce il vettore di attacco, migliorando il profilo di sicurezza dell'organizzazione.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Automation

Nel definire una strategia di implementazione dell'infrastruttura immutabile, si consiglia di utilizzare il più possibile l'automazione per aumentare la riproducibilità e ridurre al minimo i potenziali errori umani. Per maggiori dettagli, consulta REL08-BP05 Implementazione delle modifiche tramite automazione e Automatizzazione di distribuzioni pratiche e sicure.

Con il modello infrastructure as code (IaC), le fasi di provisioning, orchestrazione e implementazione dell'infrastruttura sono definite in modo programmatico, descrittivo e dichiarativo, e conservate in un sistema di controllo del codice sorgente. L'utilizzo del modello Infrastructure as code (IaC) semplifica l'automazione dell'implementazione dell'infrastruttura e aiuta a raggiungere l'immutabilità dell'infrastruttura.

Modelli di implementazione

Quando è richiesta una modifica del carico di lavoro, la strategia di implementazione immutabile dell'infrastruttura impone l'implementazione di un nuovo set di risorse dell'infrastruttura, comprese tutte le modifiche necessarie. È importante che questo nuovo set di risorse si basi su un modello di implementazione che riduca al minimo l'impatto sugli utenti. Esistono due strategie principali per questa implementazione:

Distribuzione canary: è la pratica di indirizzare un piccolo numero di clienti alla nuova versione, in genere in esecuzione su una singola istanza di servizio (la release canary). Quindi analizzerai in modo approfondito le modifiche di comportamento o gli errori generati. Puoi rimuovere il traffico dalla release canary in caso di problemi critici e reindirizzare gli utenti alla versione precedente. Se l'implementazione viene completata correttamente, puoi continuare a implementare alla velocità

desiderata, monitorando le modifiche alla ricerca di errori, fino a quando l'implementazione non sarà completata. AWS CodeDeploy può essere configurato con una configurazione di implementazione che consente una distribuzione canary.

Implementazione blu/verde: simile alla distribuzione canary, tranne per il fatto che un intero parco dell'applicazione è implementato in parallelo. Puoi alternare le implementazioni tra i due stack (blu e verde). Ancora una volta, puoi inviare il traffico alla nuova versione e tornare alla versione precedente in caso di problemi con l'implementazione. Generalmente, tutto il traffico viene trasferito contemporaneamente, tuttavia puoi anche utilizzare frazioni del traffico verso ciascuna versione per accelerare l'adozione della nuova versione mediante le funzionalità di instradamento DNS ponderato di Amazon Route 53. AWS CodeDeploy e AWS Elastic Beanstalk possono essere impostati con una configurazione di implementazione che consente un'implementazione blu/verde.

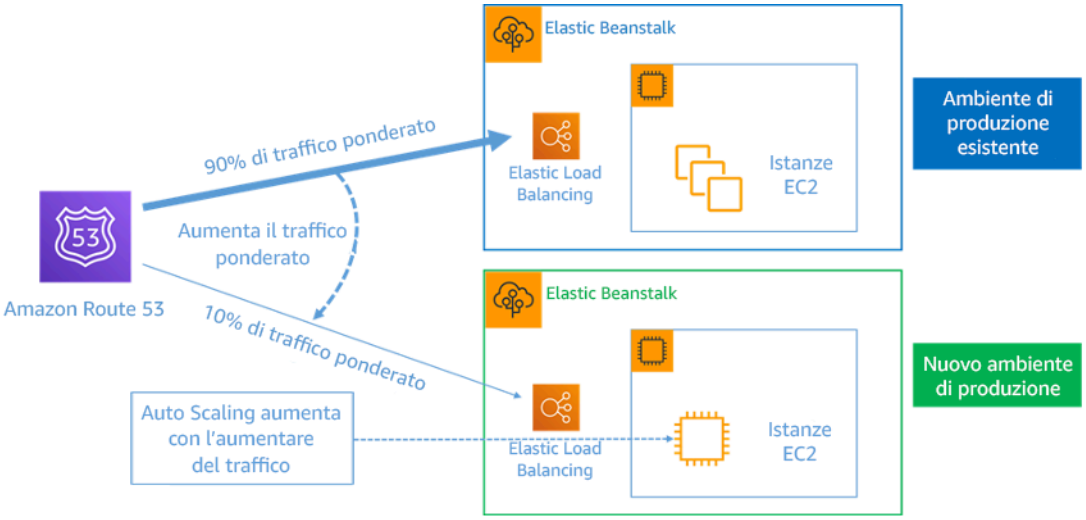


Figura 8: Implementazione blu/verde con AWS Elastic Beanstalk e Amazon Route 53

Rilevamento delle deviazioni

Per deviazione si intende qualsiasi modifica che causa uno stato o una configurazione di una risorsa dell'infrastruttura diversi da quelli previsti. Qualsiasi tipo di modifica non gestita della configurazione è contraria al concetto di infrastruttura immutabile e tale modifica dovrebbe essere individuata e corretta per implementare con successo l'infrastruttura immutabile.

Passaggi dell'implementazione

- Non autorizzare la modifica locale (in-place) delle risorse dell'infrastruttura in esecuzione.
  - Puoi usare AWS Identity and Access Management (IAM) per specificare chi o cosa può accedere a servizi e risorse in AWS, gestire a livello centrale le autorizzazioni in modo granulare e analizzare l'accesso per perfezionare le autorizzazioni in AWS.

- Automatizza l'implementazione delle risorse dell'infrastruttura per aumentare la riproducibilità e ridurre al minimo i potenziali errori umani.
  - Come illustrato nel whitepaper Introduzione a DevOps in AWS, l'automazione è fondamentale per i servizi AWS ed è supportata a livello interno in tutti i servizi, le funzionalità e le offerte.
  - La preparazione preliminare di Amazon Machine Image (AMI) può velocizzare i tempi di avvio. EC2 Image Builder è un servizio AWS completamente gestito che consente di automatizzare creazione, manutenzione, convalida, condivisione e implementazione di AMI personalizzate, sicure e aggiornate per Linux o Windows.
  - Alcuni dei servizi che supportano l'automazione sono:
    - AWS Elastic Beanstalk è un servizio per implementare e scalare rapidamente applicazioni e servizi Web sviluppati con Java, .NET, PHP, Node.js, Python, Ruby, Go e Docker su server comuni come Apache, NGINX, Passenger e IIS.
    - AWS Proton consente ai team della piattaforma di connettere e coordinare tutti i vari strumenti necessari ai team di sviluppo per il provisioning dell'infrastruttura, l'implementazione del codice, il monitoraggio e gli aggiornamenti. AWS Proton abilita il provisioning e l'implementazione basati sul modello Infrastructure as code di applicazioni serverless e basate su container.
  - L'utilizzo del modello Infrastructure as code (IaC) semplifica l'automazione dell'implementazione dell'infrastruttura e aiuta a raggiungere l'immutabilità dell'infrastruttura. AWS fornisce servizi che consentono la creazione, l'implementazione e la manutenzione dell'infrastruttura in modo programmatico, descrittivo e dichiarativo.
    - AWS CloudFormation consente agli sviluppatori di creare risorse AWS in modo ordinato e prevedibile. Le risorse sono scritte in file di testo utilizzando il formato JSON o YAML. I modelli richiedono una sintassi e una struttura specifiche che dipendono dai tipi di risorse create e gestite. Crei le risorse in formato JSON o YAML con qualsiasi editor di codice e le inserisci in un sistema di controllo delle versioni. A questo punto, CloudFormation crea i servizi specificati in modo sicuro e ripetibile.
    - AWS Serverless Application Model (AWS SAM) è un framework open source utilizzabile per la creazione di applicazioni serverless in AWS. AWS SAM si integra con altri servizi AWS, oltre a essere un'estensione di CloudFormation.
    - AWS Cloud Development Kit (AWS CDK) è un framework di sviluppo software open source per modellare ed eseguire il provisioning delle risorse delle applicazioni cloud utilizzando linguaggi di programmazione familiari. È possibile utilizzare AWS CDK per modellare l'infrastruttura dell'applicazione mediante TypeScript, Python, Java e.NET. AWS CDK utilizza CloudFormation in background per fornire risorse in modo sicuro e ripetibile.

    - AWS Cloud Control API introduce un set comune di API Create, Read, Update, Delete e List (CRUDL) per consentire agli sviluppatori di gestire la propria infrastruttura cloud in modo semplice e coerente. Le API comuni (API Cloud Control) consentono agli sviluppatori di gestire in modo uniforme il ciclo di vita di AWS e i servizi di terze parti.
- Applica modelli di implementazione che riducano al minimo l'impatto sugli utenti.
    - Distribuzione canary:
        - Set up an API Gateway canary release deployment
        - Create a pipeline with canary deployments for Amazon ECS using AWS App Mesh
    - Implementazioni blu/verde: il whitepaper Blue/Green Deployments on AWS riporta tecniche esemplificative per implementare strategie di implementazione blu/verde.
- Rileva le deviazioni a livello di configurazione o stato. Per ulteriori informazioni, consulta Detecting unmanaged configuration changes to stacks and resources.

Risorse

Best practice correlate:

- REL08-BP05 Implementazione delle modifiche tramite automazione

Documenti correlati:

- Automatizzazione di distribuzioni pratiche e sicure
- Leveraging AWS CloudFormation to create an immutable infrastructure at Nubank
- Infrastructure as code
- Implementing an alarm to automatically detect drift in AWS CloudFormation stacks

Video correlati:

- AWS re:Invent 2020: Reliability, consistency, and confidence through immutability

REL08-BP05 Implementazione delle modifiche tramite automazione

Le implementazioni e l'applicazione di patch sono automatizzate per eliminare l'impatto negativo.

Apportare modifiche ai sistemi produttivi è una delle maggiori aree di rischio per molte organizzazioni. Riteniamo che le implementazioni siano un problema prioritario da risolvere insieme ai problemi

aziendali affrontati dal software. Oggi, ciò significa l'uso dell'automazione ovunque sia pratica nelle operazioni, inclusi test e implementazione di modifiche, aggiunta o rimozione di capacità e migrazione dei dati.

Risultato desiderato: integrazione della sicurezza dell'implementazione automatizzata nel processo di rilascio con test di pre-produzione completi, rollback automatici e implementazioni di produzione scaglionate. Questa automazione riduce al minimo il potenziale impatto sulla produzione causato da implementazioni non riuscite e gli sviluppatori non devono più monitorare attivamente le implementazioni in produzione.

Anti-pattern comuni:

- Esegui le modifiche manualmente.
- Non esegui le fasi nell'automazione tramite flussi di lavoro manuali di emergenza.
- Non segui i piani e i processi stabiliti a favore di tempistiche accelerate.
- Esegui implementazioni successive rapide senza attendere il tempo di incorporamento.

Vantaggi dell'adozione di questa best practice: l'utilizzo dell'automazione per implementare tutte le modifiche elimina la possibilità di introdurre errori umani, oltre a offrire la possibilità di eseguire test prima di apportare modifiche alla produzione. L'esecuzione di questo processo prima del passaggio in produzione verifica che i piani siano completi. Inoltre, il rollback automatico del processo di rilascio può identificare i problemi di produzione e riportare il carico di lavoro allo stato operativo precedente.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Automatizzazione della pipeline di implementazione Le pipeline di implementazione permettono di richiamare test automatici, rilevare le anomalie e interrompere la pipeline a una determinata fase prima dell'implementazione in produzione o eseguire automaticamente il ripristino di una modifica. Parte integrante di ciò è l'adozione della cultura basata sull'integrazione continua e sulla consegna/implementazione continua (CI/CD), dove un commit o una modifica del codice passa lungo varie fasi automatizzate, dalle fasi di creazione e test, fino all'implementazione negli ambienti di produzione.

Anche se la prassi comune suggerisce di includere le persone nelle procedure operative più difficili, suggeriamo di automatizzare le procedure più difficili proprio per questo motivo.

Passaggi dell'implementazione

Per automatizzare le implementazioni ed eliminare le operazioni manuali, segui questi passaggi:

- Configura un repository di codice per conservare il codice in modo sicuro: utilizza un sistema di gestione del codice sorgente ospitato basato su una tecnologia popolare come Git per memorizzare il codice sorgente e la configurazione infrastructure as code (IaC).
- Configura un servizio di integrazione continua per compilare il codice sorgente, eseguire test e creare artefatti di implementazione: per configurare un progetto di compilazione a tale scopo, consulta Getting started with AWS CodeBuild using the console.
- Configura un servizio di implementazione in grado di automatizzare le implementazioni delle applicazioni e gestire la complessità degli aggiornamenti delle stesse senza fare affidamento su implementazioni manuali soggette a errori: AWS CodeDeploy automatizza le implementazioni software in svariati servizi di calcolo, come Amazon EC2, AWS Fargate, AWS Lambda e i tuoi server on-premises. Per la configurazione di questi passaggi, consulta Getting started with CodeDeploy.
- Imposta un servizio di distribuzione continua in grado di automatizzare le pipeline di rilascio per aggiornamenti più rapidi e affidabili delle applicazioni e dell'infrastruttura: prendi in considerazione l'utilizzo di AWS CodePipeline per automatizzare le tue pipeline di rilascio. Per maggiori dettagli, consulta CodePipeline tutorials.

Risorse

Best practice correlate:

- OPS05-BP04 Utilizzo di sistemi di gestione della compilazione e implementazione
- OPS05-BP10 Automazione completa dell'integrazione e dell'implementazione
- OPS06-BP02 Implementazioni dei test
- OPS06-BP04 Automazione dei test e del rollback

Documenti correlati:

- Continuous Delivery of Nested AWS CloudFormation Stacks Using AWS CodePipeline
- Partner APN: partner per la creazione di soluzioni di implementazione automatizzate
- Marketplace AWS: prodotti per l'automazione delle implementazioni
- Automatizza i messaggi delle chat con webhook.
- Amazon Builders' Library: garantire la sicurezza del rollback durante le distribuzioni
- Amazon Builders' Library: più velocità con una consegna continua

- Cosa è AWS CodePipeline?
- What Is CodeDeploy?
- AWS Systems Manager Patch Manager
- What is Amazon SES?
- What is Amazon Simple Notification Service?

Video correlati:

- AWS Summit 2019: CI/CD on AWS

Gestione dei guasti

Questions

- REL 9. In che modo eseguire il backup dei dati?
- REL 10. Come si utilizza l'isolamento dei guasti per proteggere il carico di lavoro?
- REL 11. Come si progetta il carico di lavoro affinché resista ai guasti dei componenti?
- REL 12. Come si testa l'affidabilità?
- REL 13. Come si pianifica il disaster recovery?

REL 9. In che modo eseguire il backup dei dati?

Esegui il backup dei dati, delle applicazioni e della configurazione per soddisfare i requisiti relativi agli obiettivi del tempo di ripristino (RTO) e agli obiettivi del punto di ripristino (RPO).

Best practice

- REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o una riproduzione dei dati dalle origini
- REL09-BP02 Protezione e crittografia dei backup
- REL09-BP03 Esecuzione del backup dei dati in automatico
- REL09-BP04 Ripristino periodico dei dati per verificare l'integrità e i processi di backup:

REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o una riproduzione dei dati dalle origini

Scopri e utilizza le funzionalità di backup dei servizi e delle risorse di dati usati dal carico di lavoro. La maggior parte dei servizi offre funzionalità per eseguire il backup dei dati del carico di lavoro.

Risultato desiderato: le origini dati sono state identificate e classificate in base alla criticità. Quindi, stabilisci una strategia per il recupero dei dati in base all'RPO. Questa strategia prevede il backup di queste origini dati o la possibilità di riprodurre i dati da altre origini. In caso di perdita di dati, la strategia implementata consente il recupero o la riproduzione dei dati entro i termini RPO e RTO definiti.

Fase di maturità del cloud: di base

Anti-pattern comuni:

- Mancata conoscenza di tutte le origini dati per il carico di lavoro e della loro criticità.
- Non si eseguono backup delle origini dati critiche.
- Esecuzione di backup solo di alcune origini dati senza utilizzare la criticità come criterio.
- Non esiste un RPO definito o la frequenza di backup non può soddisfare l'RPO.
- Nessuna valutazione della necessità di un backup o della possibilità di riprodurre i dati da altre origini.

Vantaggi dell'adozione di questa best practice: l'identificazione dei punti in cui sono necessari i backup e l'implementazione di un meccanismo per la creazione di backup, o la possibilità di riprodurre i dati da una fonte esterna, migliorano la capacità di ripristinare e recuperare i dati durante un'interruzione.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Tutti i data store AWS offrono funzionalità di backup. Servizi come Amazon RDS e Amazon DynamoDB supportano inoltre il backup automatico che consente il recupero point-in-time (PITR), grazie al quale è possibile ripristinare un backup in qualsiasi momento fino a cinque minuti o meno rispetto all'ora corrente. Diversi servizi AWS offrono la possibilità di copiare i backup su un altro Regione AWS. AWS Backup è uno strumento che permette di centralizzare e automatizzare la protezione dei dati tra i servizi AWS. AWS Elastic Disaster Recovery consente di copiare carichi di

lavoro server completi e mantenere una protezione continua dei dati on-premises, tra diverse zone di disponibilità o tra regioni con un Obiettivo del punto di ripristino (RPO) misurato in secondi.

Amazon S3 può essere utilizzato come destinazione di backup per le origini dati gestite dal cliente e da AWS. I servizi AWS come Amazon EBS, Amazon RDS, e Amazon DynamoDB presentano funzionalità integrate per la creazione di backup. È anche possibile utilizzare software di backup di terze parti.

È possibile eseguire il backup dei dati on-premises in Cloud AWS utilizzando Gateway di archiviazione AWS o AWS DataSync. È possibile usare i bucket Amazon S3 per archiviare questi dati in AWS. Amazon S3 offre più livelli di archiviazione come Amazon Glacier o Amazon Glacier Deep Archive per ridurre i costi dell'archiviazione di dati.

Potresti essere in grado di soddisfare le esigenze di recupero dei dati riproducendo i dati da altre origini. Ad esempio, i nodi di replica di Amazon ElastiCache o le repliche di lettura di Amazon RDS consentono di riprodurre i dati in caso di perdita del nodo primario. In caso di possibile utilizzo di queste origini per soddisfare l'Obiettivo del punto di ripristino (RPO) e l'Obiettivo del tempo di ripristino (RTO), potrebbe non essere necessario un backup. Un altro esempio: con Amazon EMR, potrebbe non essere necessario eseguire il backup del data store HDFS, finché è possibile riprodurre i dati in Amazon EMR da Amazon S3.

Quando scegli una strategia di backup, devi considerare il tempo necessario per il ripristino dei dati. Il tempo necessario per il ripristino dei dati dipende dal tipo di backup (nel caso di una strategia di backup) o dalla complessità del meccanismo di riproduzione dei dati. Questo tempo deve rientrare nell'RTO per il carico di lavoro.

Passaggi dell'implementazione

1. Identifica tutte le origini dati per il carico di lavoro. L'archiviazione dei dati può avvenire su varie risorse come database, volumi, file system, sistemi di log e storage a oggetti. Consulta la sezione Risorse per trovare i documenti correlati in merito ai vari servizi AWS di archiviazione dei dati e alle funzionalità di backup fornite da questi.
2. Classifica le origini dati in base alla criticità. I diversi set di dati avranno diversi livelli di criticità per un carico di lavoro e quindi diversi requisiti di resilienza. Ad esempio, alcuni dati possono essere critici e richiedere un RPO prossimo allo zero, mentre altri dati possono essere meno critici e tollerare un RPO più elevato e una certa perdita di dati. Allo stesso modo, anche i diversi set di dati possono avere requisiti RTO diversi.
3. Utilizza i servizi AWS o di terze parti per creare backup dei dati. AWS Backup è un servizio gestito che consente la creazione di backup di varie origini dati su AWS. AWS Elastic Disaster Recovery

gestisce la replica automatizzata dei dati in meno di un secondo in una Regione AWS. La maggior parte dei servizi AWS include anche funzionalità native per la creazione di backup. Marketplace AWS offre molte soluzioni che offrono anche queste funzionalità. Consulta la sezione Risorse più avanti per informazioni su come creare backup dei dati da vari servizi AWS.

4. Per i dati non sottoposti a backup, definisci un meccanismo di riproduzione dei dati. Puoi decidere di non eseguire il backup di dati riproducibili da altre origini per vari motivi. Potrebbe essere più conveniente riprodurre i dati dalle origini, quando necessario, piuttosto che creare un backup, dato che l'archiviazione dei backup può comportare dei costi. Un altro esempio è quello in cui il ripristino da un backup richiede più tempo rispetto alla riproduzione dei dati dalle origini, con conseguente violazione dell'RTO. In queste situazioni, è necessario considerare i compromessi e stabilire un processo ben definito per la riproduzione dei dati da queste origini quando è necessario il ripristino dei dati. Ad esempio, se hai caricato dati da Amazon S3 a un data warehouse (ad esempio Amazon Redshift) o a un cluster MapReduce (ad esempio Amazon EMR) per eseguire analisi su tali dati, questo può essere un esempio di dati che possono essere riprodotti da altre origini. Finché i risultati di queste analisi vengono archiviati o sono riproducibili, non subirai una perdita di dati a causa di un guasto nel data warehouse o nel cluster MapReduce. Altri esempi che possono essere riprodotti dalle origini includono le cache (ad esempio Amazon ElastiCache) o le repliche di lettura RDS.
5. Definisci una cadenza per il backup dei dati. La creazione di backup delle origini dei dati è un processo periodico e la frequenza deve dipendere dall'RPO.

Livello di impegno per il piano di implementazione: moderato

Risorse

Best practice correlate:

REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati

REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino

Documenti correlati:

- Cosa è AWS Backup?
- In cosa consiste AWS DataSync?
- Cosa si intende per Gateway di volumi?
- Partner APN: partner che possono aiutare con il backup

- Marketplace AWS: prodotti che possono essere utilizzati per il backup
- Snapshot Amazon EBS
- Backup di Amazon EFS
- Backup di Amazon FSx per Windows File Server
- Backup e ripristino per ElastiCache for Redis
- Creazione di un DB Cluster Snapshot in Neptune
- Creazione di un DB Snapshot
- Creazione di una regola EventBridge che si attiva in base a un piano
- Replica tra regioni con Amazon S3
- EFS-to-EFS AWS Backup
- Esportazione dei dati di log su Amazon S3
- Gestione del ciclo di vita degli oggetti
- Backup e ripristino on demand per DynamoDB
- Ripristino point-in-time per DynamoDB
- Uso di snapshot di indici del servizio OpenSearch di Amazon
- In cosa consiste AWS Elastic Disaster Recovery?

Video correlati:

- AWS re:Invent 2021 - Backup, disaster recovery, and ransomware protection with AWS
- AWS Backup Demo: Cross-Account and Cross-Region Backup
- AWS re:Invent 2019: Deep dive on AWS Backup, ft. Rackspace (STG341)

REL09-BP02 Protezione e crittografia dei backup

Controlla e rileva l'accesso ai backup utilizzando l'autenticazione e l'autorizzazione. Previeni e rileva se l'integrità dei dati dei backup è compromessa utilizzando la crittografia.

Implementa controlli di sicurezza per impedire l'accesso non autorizzato ai dati di backup. Esegui la crittografia dei backup per proteggere la riservatezza e l'integrità dei dati.

Anti-pattern comuni:

- Disporre di un accesso identico sia per i backup e l'automazione del ripristino sia per i dati.
- Non codificare i backup.
- Non implementare l'immutabilità per la protezione da cancellazioni o manomissioni.
- Utilizzo dello stesso dominio di sicurezza per i sistemi di produzione e di backup.
- Non convalidare l'integrità del backup mediante test regolari.

Vantaggi dell'adozione di questa best practice:

- La protezione dei backup previene la manomissione dei dati, mentre la crittografia dei dati impedisce l'accesso in caso di esposizione accidentale.
- Protezione avanzata contro il ransomware e altre minacce informatiche che prendono di mira l'infrastruttura di backup.
- Tempi di ripristino ridotti a seguito di un incidente informatico grazie a processi di ripristino convalidati.
- Funzionalità di continuità aziendale migliorate durante gli incidenti di sicurezza.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Controlla e rileva l'accesso ai backup tramite l'autenticazione e l'autorizzazione, ad esempio con AWS Identity and Access Management (IAM). Previeni e rileva se l'integrità dei dati dei backup è compromessa utilizzando la crittografia.

Amazon S3 supporta diversi metodi di crittografia dei dati a riposo. Utilizzando la crittografia lato server, Amazon S3 accetta anche dati non crittografati e li crittografa man mano che vengono memorizzati. Utilizzando la crittografia lato client, l'applicazione del carico di lavoro è responsabile della crittografia dei dati prima che vengano inviati ad Amazon S3. Entrambi i metodi ti consentono di utilizzare AWS Key Management Service (AWS KMS) per creare ed archiviare la chiave di crittografia dei dati, oppure di utilizzarne una personalizzata (della quale sarai responsabile). Tramite AWS KMS, puoi impostare delle policy utilizzando IAM per regolare l'accesso alle chiavi dei dati, oltre che ai dati privi di crittografia.

Per Amazon RDS, se hai scelto di crittografare i database, anche i backup vengono crittografati. I backup di DynamoDB sono sempre crittografati. Quando usi AWS Elastic Disaster Recovery, vengono crittografati tutti i dati in transito e a riposo. Con Elastic Disaster Recovery, i dati a riposo

possono essere crittografati tramite la chiave di crittografia dei volumi della crittografia predefinita in Amazon EBS o una chiave gestita dal cliente personalizzata.

Considerazioni sulla resilienza informatica

Per migliorare la sicurezza del backup contro le minacce informatiche, prendi in considerazione l'implementazione di questi controlli aggiuntivi oltre alla crittografia:

- Implementa l'immutabilità utilizzando AWS Backup Vault Lock o Amazon S3 Object Lock per impedire che i dati di backup vengano alterati o eliminati durante il periodo di conservazione, proteggendoli dal ransomware e dall'eliminazione intenzionale.
- Stabilisci l'isolamento logico tra gli ambienti di produzione e di backup con vault AWS Backup logicamente isolata per i sistemi critici, creando una separazione che aiuta a prevenire la compromissione simultanea di entrambi gli ambienti.
- Convalida regolarmente l'integrità del backup utilizzando i test di ripristino AWS Backup per verificare che i backup non siano danneggiati e possano essere ripristinati con successo a seguito di un incidente informatico.
- Implementa l'approvazione multiparte per le operazioni di ripristino critiche utilizzando l'approvazione AWS Backup multipartitica per prevenire tentativi di ripristino non autorizzati o dannosi richiedendo l'autorizzazione di più approvatori designati.

Passaggi dell'implementazione

1. Utilizzo della crittografia su ciascuno dei datastore. Se i dati di origine sono crittografati, lo sarà anche il backup.
   - Utilizza la crittografia in Amazon RDS. Puoi configurare la crittografia dei dati a riposo utilizzando AWS Key Management Service al momento della creazione di un'istanza RDS.
   - Utilizza la crittografia sui volumi Amazon EBS. Puoi configurare la crittografia predefinita o specificare una chiave univoca al momento della creazione del volume.
   - Utilizza la crittografia di Amazon DynamoDB necessaria. DynamoDB codifica tutti i dati a riposo. Puoi utilizzare una chiave AWS KMS di proprietà di AWS o una chiave KMS gestita da AWS specificando una chiave archiviata nel tuo account.
   - Codifica i dati archiviati in Amazon EFS. Configura la crittografia al momento della creazione del file system.
   - Configura la crittografia nelle regioni di origine e di destinazione. Puoi configurare la crittografia dei dati a riposo in Amazon S3 utilizzando le chiavi archiviate in KMS tenendo presente che le

chiavi sono specifiche per regione. Puoi specificare le chiavi di destinazione quando configuri la replica.

- Scegli se utilizzare la crittografia Amazon EBS per Elastic Disaster Recovery predefinita o personalizzata. Questa opzione esegue la crittografia dei dati a riposo replicati nei dischi della sottorete dell'area di staging e nei dischi replicati.

2. Implementazione delle autorizzazioni con privilegio minimo per accedere ai backup. Segui le best practice per limitare l'accesso a backup, snapshot e repliche in conformità con le best practice di sicurezza.
3. Configura l'immutabilità per i backup critici. Per i dati critici, implementa AWS Backup Vault Lock o S3 Object Lock per impedire l'eliminazione o l'alterazione durante il periodo di conservazione specificato. Per informazioni sull'implementazione, consulta AWS Backup Vault Lock.
4. Crea una separazione logica per gli ambienti di backup. Implementa una vault logicamente isolata AWS Backup per i sistemi critici che richiedono una protezione avanzata dalle minacce informatiche. Per una guida all'implementazione, consulta Building cyber resiliency with AWS Backup logically air-gapped vault.
5. Implementa i processi di convalida del backup. Configura i test di ripristino AWS Backup per verificare regolarmente che i backup non siano danneggiati e possano essere ripristinati correttamente a seguito di un incidente informatico. Per ulteriori informazioni, consulta Convalidare la preparazione al ripristino con i test di ripristino AWS Backup.
6. Configura l'approvazione multiparte per le operazioni di ripristino sensibili. Per i sistemi critici, implementa l'approvazione multiparte AWS Backup per richiedere l'autorizzazione di più approvatori designati prima di procedere con il ripristino. Per i dettagli sull'implementazione, consulta Migliorare la resilienza del ripristino con il supporto AWS Backup per l'approvazione multiparte.

Risorse

Documenti correlati:

- Marketplace AWS: prodotti che possono essere utilizzati per il backup
- Amazon EBS Encryption
- Amazon S3: protezione dei dati tramite la crittografia
- Configurazione aggiuntiva CRR: replica di oggetti creati con crittografia lato server (SSE) utilizzando le chiavi di crittografia archiviate in AWS KMS
- DynamoDB Encryption at Rest

- Encrypting Amazon RDS Resources
- Crittografia dei dati e dei metadati in Amazon EFS
- Encryption for Backups in AWS
- Gestione di tabelle crittografate
- Pilastro della sicurezza: Framework AWS Well-Architected
- In cosa consiste AWS Elastic Disaster Recovery?
- FSISEC11: Come vi proteggete dal ransomware?
- Gestione del rischio di ransomware su AWS Uso del framework di sicurezza informatica NIST
- Creazione della resilienza informatica tramite una vault logicamente isolata AWS Backup
- Convalida la preparazione al ripristino con i test di ripristino AWS Backup
- Migliora la resilienza del ripristino con il supporto AWS Backup per l'approvazione multiparte

Esempi correlati:

- Well-Architected Lab: implementazione della replica bidirezionale tra regioni per Amazon S3

REL09-BP03 Esecuzione del backup dei dati in automatico

Configura i backup in modo che vengano eseguiti automaticamente in base a una pianificazione periodica informata dall'Obiettivo del punto di ripristino (RPO) o dalle modifiche apportate al set di dati. I set di dati critici con bassi requisiti di perdita di dati devono essere sottoposti a backup automatico su base frequente, mentre i dati meno critici, per i quali è accettabile una certa perdita, possono essere sottoposti a backup meno frequenti.

Risultato desiderato: un processo automatizzato che crea backup delle origini dati con una cadenza stabilita.

Anti-pattern comuni:

- Eseguire i backup manualmente.
- Utilizzare risorse che dispongono di funzionalità di backup, ma non includere il backup nell'automazione.

Vantaggi dell'adozione di questa best practice: l'automazione dei backup verifica che vengano eseguiti regolarmente in base all'RPO e avvisa se non vengono eseguiti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

AWS Backup consente di creare backup automatici di dati di varie origini dati AWS. Il backup delle istanze Amazon RDS può essere eseguito quasi ininterrottamente ogni cinque minuti e quello degli oggetti Amazon S3 quasi ininterrottamente ogni quindici minuti, consentendo il ripristino point-in-time (PITR) a un punto specifico della cronologia di backup. Per altre origini dati AWS, come volumi Amazon EBS, tabelle Amazon DynamoDB o file system Amazon FSx, AWS Backup può eseguire il backup automatico con una frequenza di un'ora. Questi servizi offrono inoltre funzionalità di backup native. I servizi AWS con backup automatizzato e ripristino point-in-time includono Amazon DynamoDB, Amazon RDS e Amazon Keyspaces (per Apache Cassandra), il cui ripristino è possibile in un momento specifico all'interno della cronologia di backup. La maggior parte degli altri servizi di archiviazione di dati AWS offre la possibilità di programmare backup periodici, anche ogni ora.

Amazon RDS e Amazon DynamoDB offrono il backup continuo con ripristino point-in-time. Il controllo delle versioni di Amazon S3, una volta abilitato, è automatico. È possibile utilizzare Amazon Data Lifecycle Manager per automatizzare la creazione, la copia e l'eliminazione di snapshot Amazon EBS. Può anche automatizzare la creazione, la copia, la rimozione e la cancellazione di Amazon Machine Image (AMI) con backup Amazon EBS e dei relativi snapshot Amazon EBS sottostanti.

AWS Elastic Disaster Recovery offre la replica a livello di blocco continua dall'ambiente di origine (on-premises o AWS) alla regione di ripristino di destinazione. Gli snapshot Amazon EBS point-in-time vengono creati e gestiti automaticamente dal servizio.

Per una visualizzazione centralizzata dell'automazione e della cronologia dei backup, AWS Backup fornisce una soluzione di backup completamente gestita basata su policy. Centralizza e automatizza il backup dei dati su più servizi AWS nel cloud e on-premises utilizzando Gateway di archiviazione AWS.

Oltre al controllo delle versioni, Amazon S3 offre la funzionalità di replica. L'intero bucket S3 può essere replicato automaticamente in un altro bucket in una Regione AWS diversa.

Passaggi dell'implementazione

1. Identifica le origini dati al momento sottoposte a backup manuale. Per ulteriori dettagli, consulta REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o una riproduzione dei dati dalle origini.
2. Determina l'RPO per il carico di lavoro. Per ulteriori dettagli, consulta REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati.

3. Utilizza una soluzione di backup automatico o un servizio gestito. AWS Backup è un servizio totalmente gestito che semplifica la centralizzazione e l'automatizzazione della protezione dei dati in tutti i servizi AWS, nel cloud e on-premises. Usando piani di backup in AWS Backup, crea regole che definiscano le risorse di cui eseguire il backup e la frequenza di creazione dei backup. Questa frequenza deve essere informata dall'RPO stabilito al punto 2. Per una guida pratica su come creare backup automatici con AWS Backup, consulta Testing Backup and Restore of Data. La maggior parte dei servizi AWS di archiviazione dei dati offre funzionalità di backup native. Ad esempio, RDS può essere sfruttato per backup automatici con ripristino point-in-time (PITR).
4. Per le origini dati non supportate da una soluzione di backup automatico o da un servizio gestito, come le origini dati on-premises o le code di messaggi, è consigliabile utilizzare una soluzione di terze parti affidabile per creare backup automatici. In alternativa, puoi creare un'automazione utilizzando la AWS CLI o gli SDK. Puoi usare funzioni AWS Lambda o AWS Step Functions per definire la logica necessaria per la creazione di un backup di dati e utilizzare Amazon EventBridge per richiamare la stessa in base a una frequenza determinata dall'RPO.

Livello di impegno per il piano di implementazione: basso

Risorse

Documenti correlati:

- Partner APN: partner per il backup
- Marketplace AWS: prodotti che possono essere utilizzati per il backup
- Creating an EventBridge Rule That Triggers on a Schedule
- Che cos'è AWS Backup?
- Che cos'è AWS Step Functions?
- What is AWS Elastic Disaster Recovery?

Video correlati:

- AWS re:Invent 2019: Deep dive on AWS Backup, ft. Rackspace (STG341)

REL09-BP04 Ripristino periodico dei dati per verificare l'integrità e i processi di backup:

Verifica che l'implementazione del processo di backup soddisfi gli obiettivi del tempo di ripristino (RTO) e gli obiettivi del punto di ripristino (RPO) eseguendo un test del ripristino.

Risultato desiderato: i dati dei backup vengono ripristinati periodicamente utilizzando meccanismi ben definiti per verificare che il ripristino sia possibile entro l'obiettivo del tempo di ripristino (RTO) stabilito per il carico di lavoro. Verifica che il ripristino da un backup porti a una risorsa che contiene i dati originali senza che questi siano danneggiati o inaccessibili e con una perdita di dati entro l'obiettivo del punto di ripristino (RPO).

Anti-pattern comuni:

- Ripristino di un backup, ma senza eseguire query sui dati o recuperarli per verificare di poter usare il ripristino.
- Presupporre l'esistenza di un backup.
- Presupporre che il backup di un sistema sia pienamente operativo e che i dati possano essere recuperati da esso.
- Presupporre che il tempo di ripristino o di recupero dei dati da un backup rientri nell'RTO del carico di lavoro.
- Presupporre che i dati contenuti nel backup rientrino nell'RPO del carico di lavoro.
- Ripristino in base alle esigenze, senza usare un runbook o seguire una procedura automatica prestabilita.

Vantaggi dell'adozione di questa best practice: il test del ripristino dei backup verifica che i dati possano essere ripristinati quando necessario senza preoccuparsi che possano essere mancanti o danneggiati, che il ripristino e il recupero siano possibili entro l'RTO per il carico di lavoro e che qualsiasi perdita di dati rientri nell'RPO per il carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

La verifica delle capacità di backup e ripristino aumenta la fiducia nella capacità di eseguire queste azioni durante un'interruzione. Ripristina periodicamente i backup in una nuova posizione ed esegui test per verificare l'integrità dei dati. Alcuni test comuni che devono essere eseguiti sono la verifica che tutti i dati siano disponibili, non siano danneggiati e siano accessibili e che un'eventuale perdita di dati rientri nell'RPO per il carico di lavoro. Questi test possono anche aiutare a verificare se i meccanismi di ripristino sono sufficientemente veloci per soddisfare l'RTO del carico di lavoro.

Con AWS, puoi creare un ambiente di test e ripristinare i backup per valutare le funzionalità RTO e RPO ed eseguire test sul contenuto e l'integrità dei dati.

Inoltre, Amazon RDS e Amazon DynamoDB consentono il ripristino point-in-time (PITR) Utilizzando il backup continuo, puoi ripristinare il set di dati allo stato in cui si trovava in una data e un'ora specificate.

Se tutti i dati sono disponibili, non sono danneggiati, sono accessibili e qualsiasi perdita di dati rientra nell'RPO del carico di lavoro. Questi test possono anche aiutare a verificare se i meccanismi di ripristino sono sufficientemente veloci per soddisfare l'RTO del carico di lavoro.

AWS Elastic Disaster Recovery offre snapshot di ripristino point-in-time (RPIT) continui di volumi Amazon EBS. Con la replica dei server di origine, gli stati point-in-time vengono registrati nel corso del tempo in base alla policy configurata. Elastic Disaster Recovery verifica l'integrità di questi snapshot avviando istanze per scopi di test ed esercitazione senza reindirizzare il traffico.

Passaggi dell'implementazione

1. Identifica le origini dati di cui stai eseguendo il backup e dove sono archiviati i backup. Per le linee guida di implementazione, consulta REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o una riproduzione dei dati dalle origini.
2. Definisci criteri per la convalida dei dati per ciascuna origine dati. Tipi di dati differenti avranno proprietà diverse che potrebbero richiedere meccanismi di convalida diversi. Considera il modo in cui potrebbero essere convalidati questi dati prima di poterli utilizzare in produzione. Alcuni modi comuni per convalidare i dati sono l'uso delle loro proprietà dei dati e del backup, come il tipo di dati, il formato, la somma di controllo, la dimensione o la combinazione di questi elementi con una logica di convalida personalizzata. Ad esempio, può trattarsi di un confronto dei valori di checksum tra la risorsa ripristinata e l'origine dati al momento della creazione del backup.
3. Definisci l'RTO e l'RPO per il ripristino dei dati in base alla relativa criticità. Per le linee guida di implementazione, consulta REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati.
4. Valuta la capacità di ripristino. Rivedi la strategia di backup e ripristino per capire se è in grado di soddisfare RTO e RPO e modifica la strategia se necessario. AWS Resilience Hub ti consente di valutare il tuo carico di lavoro. La valutazione esamina la configurazione dell'applicazione rispetto alle policy sulla resilienza e indica se gli obiettivi RTO e RPO possono essere raggiunti.
5. Esegui un ripristino di test utilizzando i processi attualmente in uso in produzione per il ripristino dei dati. Questi processi dipendono dal modo in cui è stato eseguito il backup dell'origine dati iniziale, dal formato e dalla posizione di archiviazione del backup stesso o dalla riproduzione dei dati da altre fonti. Ad esempio, in caso di utilizzo di un servizio gestito, come AWS Backup,

potrebbe essere semplice ripristinare il backup in una nuova risorsa. In caso di utilizzo di AWS Elastic Disaster Recovery, è possibile avviare un'esercitazione di ripristino.

6. Convalida il ripristino dei dati dalla risorsa ripristinata in base ai criteri stabiliti in precedenza per la convalida dei dati. I dati ripristinati e recuperati contengono il record o la voce più recente al momento del backup? Questi dati rientrano nell'RPO per il carico di lavoro?
7. Misura il tempo necessario per il recupero e ripristino, quindi confrontalo con l'RTO stabilito. Questo tempo deve rientrare nell'RTO per il carico di lavoro? Ad esempio, confronta i timestamp dell'inizio del processo di ripristino e del completamento della convalida del ripristino per calcolare la durata del processo. Tutte le chiamate API AWS hanno una datazione temporale e queste informazioni sono disponibili in AWS CloudTrail. Sebbene queste informazioni possano fornire dettagli sull'inizio del processo di ripristino, la logica di convalida dovrebbe registrare il timestamp finale del completamento della convalida. Se utilizzi un processo automatizzato, puoi sfruttare servizi come Amazon DynamoDB per archiviare queste informazioni. Inoltre, molti servizi AWS offrono una cronologia degli eventi che fornisce informazioni con data e ora in cui si sono verificate determinate azioni. All'interno di AWS Backup, le azioni di backup e di ripristino sono denominate processi. Questi contengono informazioni sulla data e l'ora come parte dei metadati che possono essere utilizzati per misurare il tempo necessario per il ripristino e il recupero.
8. Comunica alle parti interessate se la convalida dei dati non riesce o se il tempo necessario per il ripristino e il recupero supera l'RTO stabilito per il carico di lavoro. Nell'implementare l'automazione a tale scopo, come in questo lab, è possibile utilizzare servizi come Amazon Simple Notification Service (Amazon SNS) per inviare notifiche push come e-mail o SMS alle parti interessate. I messaggi in questione possono essere pubblicati anche su applicazioni di messaggistica come Amazon Chime, Slack o Microsoft Teams o utilizzati per creare attività come OpsItems mediante AWS Systems Manager OpsCenter.
9. Automatizza questo processo per eseguirlo periodicamente. Ad esempio, per automatizzare i processi di ripristino e recupero si possono utilizzare servizi come AWS Lambda o una State Machine in AWS Step Functions, mentre Amazon EventBridge può essere utilizzato per richiamare periodicamente questo flusso di lavoro di automazione, come mostrato nel diagramma di architettura sottostante. Per ulteriori informazioni, consulta Automate data recovery validation with AWS Backup. Inoltre, questo Well-Architected lab fornisce un'esperienza pratica su come realizzare l'automazione di alcuni dei passaggi qui descritti.

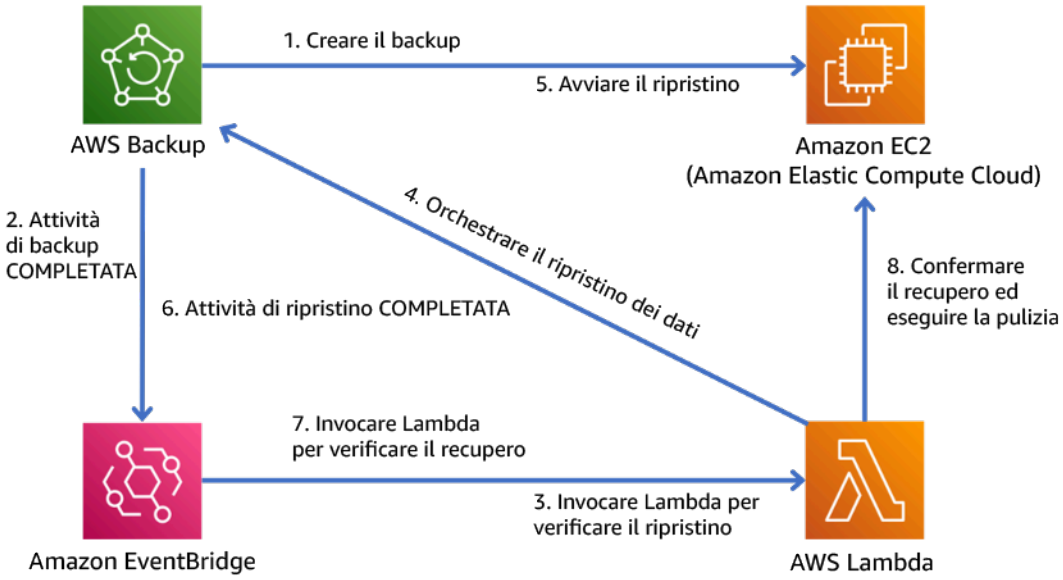


Figura 9. Processo di backup e ripristino automatico

Livello di impegno per il piano di implementazione: da moderato a elevato, in base alla complessità dei criteri di convalida.

Risorse

Documenti correlati:

- Automate data recovery validation with AWS Backup.
- Partner APN: partner per il backup
- Marketplace AWS: prodotti che possono essere utilizzati per il backup
- Creating an EventBridge Rule That Triggers on a Schedule
- Backup e ripristino on demand per DynamoDB
- Cosa è AWS Backup?
- Cosa è AWS Step Functions?
- Cos'è AWS Elastic Disaster Recovery
- AWS Elastic Disaster Recovery

REL 10. Come si utilizza l'isolamento dei guasti per proteggere il carico di lavoro?

L'isolamento dei guasti limita l'impatto di un guasto di un componente o di un sistema entro una determinata barriera. Con un isolamento adeguato, i componenti al di fuori della barriera non

subiscono gli effetti del guasto. Utilizzando più barriere per l'isolamento dei guasti, è possibile rendere un carico di lavoro più resiliente ai guasti.

Best practice

- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL10-BP02 Ripristino automatico dei componenti vincolati a una singola posizione
- REL10-BP03 Utilizzo di architetture a scomparti per limitare la portata dell'impatto

REL10-BP01 Implementazione del carico di lavoro in diversi luoghi

Distribuisci i dati e le risorse del carico di lavoro su più zone di disponibilità o, se necessario, in tutte le Regioni AWS.

Un principio fondamentale per la progettazione dei servizi in AWS è quello di evitare singoli punti di errore, inclusa l'infrastruttura fisica sottostante. AWS fornisce risorse e servizi di cloud computing a livello globale in più posizioni geografiche chiamate Regioni. Ogni Regione è fisicamente e logicamente indipendente ed è costituita da tre o più zone di disponibilità (AZ). Le zone di disponibilità sono geograficamente vicine ma fisicamente separate e isolate. Distribuendo i carichi di lavoro tra le zone di disponibilità e le Regioni, si riducono i rischi legati a minacce quali incendi, inondazioni, disastri meteorologici, terremoti ed errori umani.

Crea una strategia di localizzazione per fornire un'alta disponibilità adeguata ai carichi di lavoro.

Risultato desiderato: i carichi di lavoro di produzione sono distribuiti tra più zone di disponibilità (AZ) o Regioni per ottenere tolleranza ai guasti e alta disponibilità.

Anti-pattern comuni:

- Il carico di lavoro di produzione esiste solo in una singola zona di disponibilità.
- Viene implementata un'architettura multiregionale quando invece un'architettura multi-AZ è in grado di soddisfare i requisiti aziendali.
- Le implementazioni o i dati vengono desincronizzati, con conseguenti deviazioni di configurazione o dati sottoreplicati.
- Non tieni conto delle dipendenze tra i componenti dell'applicazione se i requisiti di resilienza e multi-posizione differiscono tra tali componenti.

Vantaggi dell'adozione di questa best practice:

- Il carico di lavoro è più resiliente in caso di incidenti, come interruzioni di corrente, problemi con i controlli ambientali, disastri naturali, errori dei servizi upstream o problemi di rete che hanno un impatto su un'AZ o su un'intera Regione.
- È possibile accedere a un inventario più ampio di istanze Amazon EC2 e ridurre le probabilità che si verifichino eccezioni InsufficientCapacityException (ICE) quando si avviano tipi specifici di istanze EC2.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Implementa e gestisci tutti i carichi di lavoro di produzione in almeno due zone di disponibilità (AZ) in una Regione.

Utilizzo di più zone di disponibilità

Le zone di disponibilità sono posizioni di hosting delle risorse fisicamente separate l'una dall'altra per evitare guasti correlati dovuti a rischi quali incendi, inondazioni e trombe d'aria. Ogni zona di disponibilità ha un'infrastruttura fisica indipendente, che include le connessioni alla rete elettrica, le fonti di alimentazione di backup, i servizi meccanici e la connettività di rete. Questa disposizione limita i guasti di uno qualsiasi di questi componenti alla sola zona di disponibilità interessata. Ad esempio, se un incidente a livello di AZ rende non disponibili le istanze EC2 nella zona di disponibilità interessata, è comunque possibile utilizzare le istanze in altre zone di disponibilità.

Nonostante siano fisicamente separate, le zone di disponibilità nella stessa Regione AWS sono sufficientemente vicine da garantire una rete a elevato throughput e bassa latenza (inferiore ai 10 millisecondi). Puoi replicare i dati in modo sincrono tra le zone di disponibilità per la maggior parte dei carichi di lavoro senza influire in modo significativo sull'esperienza dell'utente. Ciò significa che puoi utilizzare le zone di disponibilità in una Regione in una configurazione attiva/attiva o attiva/in standby.

Tutta l'elaborazione associata al carico di lavoro deve essere distribuita tra più zone di disponibilità. Sono incluse le istanze Amazon EC2, le attività AWS Fargate e le funzioni AWS Lambda collegate al VPC. I servizi di elaborazione AWS, compresi EC2 Auto Scaling, Amazon Elastic Container Service (ECS) e Amazon Elastic Kubernetes Service (EKS), offrono la possibilità di avviare e gestire l'elaborazione nelle zone di disponibilità. Configurarli per sostituire automaticamente l'elaborazione, secondo necessità, in una zona di disponibilità diversa per mantenere la disponibilità. Per indirizzare il traffico verso zone di disponibilità integre, posiziona un bilanciatore del carico davanti al computer, ad esempio un Application Load Balancer o un Network Load Balancer. I bilanciatori del carico AWS

possono reindirizzare il traffico verso le istanze disponibili in caso di compromissione della zona di disponibilità.

È inoltre necessario replicare i dati per il carico di lavoro e renderli disponibili in più zone di disponibilità. Alcuni servizi di dati gestiti da AWS, come Amazon S3, Amazon Elastic File Service (EFS), Amazon Aurora, Amazon DynamoDB, Amazon Simple Queue Service (SQS) e Flusso di dati Amazon Kinesis replicano i dati in più zone di disponibilità per impostazione predefinita e sono robusti rispetto alla compromissione della zona di disponibilità. Con altri servizi dati gestiti da AWS, come Amazon Relational Database Service (RDS), Amazon Redshift e Amazon ElastiCache, è necessario abilitare la replica multi-AZ. Una volta abilitati, questi servizi rilevano automaticamente la compromissione di una zona di disponibilità, reindirizzano le richieste verso una zona di disponibilità integra e replicano i dati in base alle esigenze dopo il ripristino senza l'intervento del cliente. Per comprendere le funzionalità, i comportamenti e le operazioni multi-AZ di ciascun servizio dati gestito da AWS utilizzato, consulta la guida per l'utente.

Se utilizzi un'archiviazione autogestita, come i volumi Amazon Elastic Block Store (EBS) o l'archiviazione di istanze Amazon EC2, devi gestire autonomamente la replica multi-AZ.

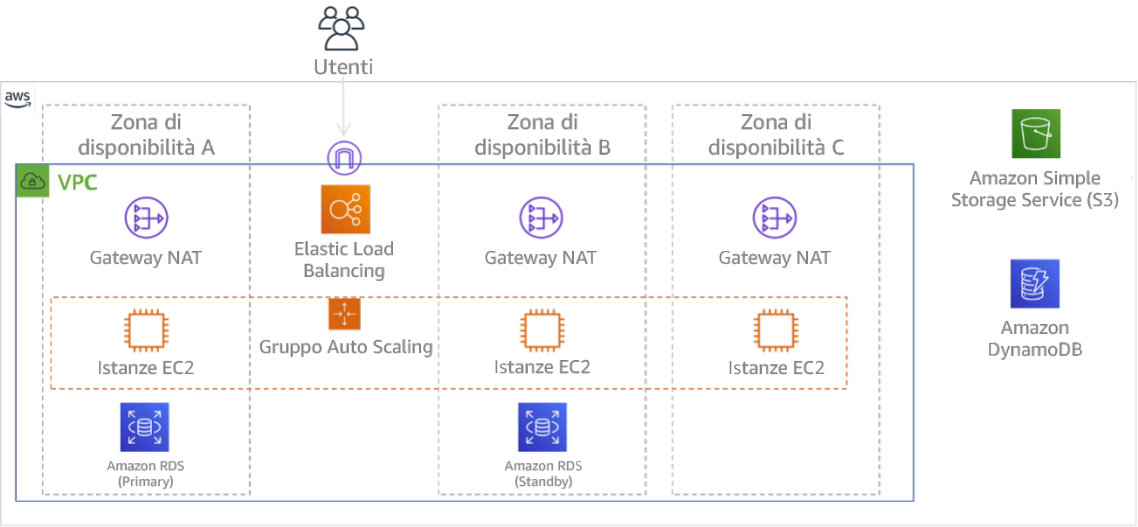


Figura 9: architettura multi-livello distribuita su tre zone di disponibilità. Nota: Amazon S3 e Amazon DynamoDB sono sempre ad AZ multiple automaticamente. L'ELB viene inoltre distribuito in tutte e tre le zone.

Utilizzo di più Regioni AWS

In presenza di carichi di lavoro che richiedono una resilienza estrema (come infrastrutture critiche, applicazioni sanitarie o servizi con requisiti di disponibilità stringenti da parte dei clienti o imposti), potrebbe essere richiesta disponibilità aggiuntiva rispetto a quella che può fornire una singola

Regione AWS. In questo caso, è necessario implementare e gestire il carico di lavoro su almeno due Regioni AWS (supponendo che ciò sia consentito dai requisiti di residenza dei dati).

Le Regioni AWS sono situate in diverse aree geografiche del mondo e in più continenti. Le Regioni AWS hanno una separazione fisica e un isolamento ancora maggiori rispetto alle sole zone di disponibilità. I servizi AWS, con poche eccezioni, sfruttano questa struttura per operare in modo completamente indipendente tra le diverse Regioni (noti anche come servizi regionali). Un guasto di un servizio in una Regione AWS, non vi è alcun impatto sul servizio in un'altra Regione.

Quando il carico di lavoro viene gestito in più Regioni, è necessario considerare ulteriori requisiti. Poiché le risorse in Regioni diverse sono separate e indipendenti l'una dall'altra, è necessario duplicare i componenti del carico di lavoro in ciascuna Regione. Questo include l'infrastruttura di base, come i VPC, oltre ai servizi dati e di elaborazione.

NOTA: se prendi in considerazione una progettazione multiregionale, verifica che sia possibile eseguire il carico di lavoro in una singola Regione. Se crei dipendenze tra le Regioni, in cui un componente di una Regione si affida a servizi o componenti di una Regione diversa, il rischio di errore potrebbe aumentare, indebolendo in maniera significativa la propria postura di affidabilità.

Per facilitare le implementazioni multiregionali e mantenere la coerenza, AWS CloudFormation StackSets può replicare l'intera infrastruttura AWS in più Regioni. AWS CloudFormation può anche rilevare deviazioni di configurazione e informare l'utente quando le risorse AWS in una Regione non sono sincronizzate. Molti servizi AWS offrono la replica in più Regioni per le risorse importanti del carico di lavoro. Ad esempio, EC2 Image Builder può pubblicare Amazon Machine Image (AMI) EC2 dopo ogni compilazione su ogni Regione utilizzata. Amazon Elastic Container Registry (ECR) può replicare le immagini del container nelle Regioni selezionate.

È inoltre necessario replicare i dati in ciascuna delle Regioni scelte. Molti servizi dati gestiti da AWS offrono funzionalità di replica multiregionale, tra cui Amazon S3, Amazon DynamoDB, Amazon RDS, Amazon Aurora, Amazon Redshift, Amazon Elasticache e Amazon EFS. Le tabelle globali di Amazon DynamoDB accettano scritture in qualsiasi Regione supportata e replicano i dati tra tutte le altre Regioni configurate. Con altri servizi, è necessario designare una Regione primaria per le scritture, mentre le altre Regioni contengono repliche di sola lettura. Per ogni servizio dati gestito da AWS utilizzato dal carico di lavoro, consulta la relativa guida per l'utente e la guida per gli sviluppatori per comprenderne le funzionalità e i limiti multiregionali. È opportuno prestare particolare attenzione a dove devono essere indirizzate le scritture, alle capacità e alle limitazioni transazionali, a come viene eseguita la replica e a come monitorare la sincronizzazione tra le Regioni.

AWS offre anche la possibilità di instradare il traffico delle richieste verso le implementazioni regionali con grande flessibilità. Ad esempio, puoi configurare i record DNS utilizzando Amazon Route 53 per indirizzare il traffico verso la Regione disponibile più vicina all'utente. In alternativa, puoi configurare i record DNS in una configurazione attiva/in standby, in cui una Regione viene designata come primaria e si ricorre a una replica regionale solo se la Regione primaria perde la propria integrità. Puoi configurare i controlli dell'integrità di Route 53 per rilevare gli endpoint non integri ed eseguire il failover automatico, nonché utilizzare Amazon Application Recovery Controller (ARC) per fornire un controllo di instradamento ad alta disponibilità per reinstradare manualmente il traffico, se necessario.

Anche se scegli di non operare in più Regioni per l'alta disponibilità, considera più Regioni come parte della propria strategia di disaster recovery (DR). Se possibile, replica i componenti e i dati dell'infrastruttura del carico di lavoro in una configurazione warm standby o fiamma pilota in una Regione secondaria. In questa progettazione, si replica l'infrastruttura di base dalla Regione primaria come VPC, gruppi Auto Scaling, orchestratori di container e altri componenti, ma si configurano i componenti di dimensioni variabili nella Regione di standby (come il numero di istanze EC2 e le repliche di database) in modo che siano di dimensioni minimamente utilizzabili. Puoi anche organizzare la replica continua dei dati dalla Regione primaria alla Regione di standby. Se si verifica un incidente, puoi aumentare orizzontalmente, o incrementare, le risorse nella Regione di standby e quindi promuoverla a Regione primaria.

Passaggi dell'implementazione

1. Collabora con le parti interessate aziendali e gli esperti in materia di residenza dei dati per determinare quali Regioni AWS possono essere utilizzate per ospitare le risorse e i dati.
2. Collabora con le parti interessate aziendali e tecniche per valutare il carico di lavoro e determinare se le esigenze di resilienza possono essere soddisfatte da un approccio multi-AZ (Regione AWS singola) o se richiedono un approccio multiregionale (se sono consentite più Regioni). L'uso di più Regioni può garantire maggiore disponibilità, ma può comportare complessità e costi aggiuntivi. Nella valutazione, considera i seguenti fattori:
    a. Obiettivi aziendali e requisiti dei clienti: quanto tempo di inattività è consentito nel caso in cui si verifichi un incidente che impatta sul carico di lavoro in una zona di disponibilità o in una Regione? Valuta gli obiettivi dei punti di ripristino come descritto in REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati.
    b. Requisiti per il disaster recovery (DR): contro quale tipo di potenziale disastro desideri assicurarti? Considera la possibilità di perdita di dati o di indisponibilità a lungo termine a livello di diversi ambiti di impatto, da una singola zona di disponibilità a un'intera Regione. Se si replicano i dati e le risorse tra le zone di disponibilità e in una singola zona di disponibilità si

verifica un guasto prolungato, il servizio può essere ripristinato in un'altra zona di disponibilità. Se si replicano i dati e le risorse tra Regioni, puoi ripristinare il servizio in un'altra Regione.

3. Distribuisci le risorse di elaborazione in più zone di disponibilità.
   a. Nel VPC, crea più sottoreti in diverse zone di disponibilità. Configura ciascuna di esse in modo che siano sufficientemente grandi da ospitare le risorse necessarie per servire il carico di lavoro, anche durante un incidente. Per ulteriori informazioni consulta REL02-BP03 Verifica che l'allocazione delle sottoreti IP consenta l'espansione e la disponibilità.
   b. Se utilizzi istanze Amazon EC2, utilizza EC2 Auto Scaling per gestire le istanze. Specifica le sottoreti scelte nel passaggio precedente durante la creazione di gruppi Auto Scaling.
   c. Se utilizzi l'elaborazione AWS Fargate per Amazon ECS o Amazon EKS, seleziona le sottoreti che hai scelto nel primo passaggio durante l'operazione di creazione di un servizio ECS, l'avvio di un'attività ECS o la creazione di un profilo Fargate per EKS.
   d. Se utilizzi funzioni AWS Lambda che devono essere eseguite nel VPC, seleziona le sottoreti che hai scelto nel primo passaggio dell'operazione di creazione della funzione Lambda. Per tutte le funzioni che non dispongono di una configurazione VPC, AWS Lambda gestisce automaticamente la disponibilità.
   e. Colloca i direttori del traffico, come i bilanciatori del carico, davanti alle risorse di elaborazione. Se il bilanciamento del carico tra zone è abilitato, AWS Application Load Balancer e Network Load Balancer rilevano quando destinazioni come istanze e container EC2 non sono raggiungibili a causa della compromissione della zona di disponibilità e reinstradano il traffico verso destinazioni in zone di disponibilità integre. Se il bilanciamento del carico tra zone è disabilitato, utilizza Amazon Application Recovery Controller (ARC) per fornire funzionalità di spostamento zonale. Se utilizzi un bilanciatore del carico di terze parti o hai implementato bilanciatori del carico personalizzati, configurali con più front-end in diverse zone di disponibilità.
4. Replica i dati del carico di lavoro in più zone di disponibilità.
   a. Se utilizzi un servizio dati gestito da AWS come Amazon RDS, Amazon ElastiCache o Amazon FSx, consulta la relativa guida per l'utente per comprendere le relative funzionalità di replica dei dati e di resilienza. Se necessario, abilita la replica e il failover tra AZ.
   b. Se utilizzi servizi di archiviazione gestiti da AWS come Amazon S3, Amazon EFS e Amazon FSx, evita di utilizzare configurazioni Single-AZ o One Zone per i dati che richiedono un'elevata durabilità. Utilizza una configurazione multi-AZ per questi servizi. Consulta la guida per l'utente del rispettivo servizio per determinare se la replica multi-AZ è abilitata per impostazione predefinita o se è necessario abilitarla.

c. Se esegui un database, una coda o un altro servizio di archiviazione autogestito, organizza la replica multi-AZ in base alle istruzioni o alle best practice dell'applicazione. Informati sulle procedure di failover della tua applicazione.

5. Configura il servizio DNS per rilevare compromissione dell'AZ e reinstrada il traffico verso una zona di disponibilità integra. Se utilizzato in combinazione con Elastic Load Balancer, Amazon Route 53 può eseguire questa operazione automaticamente. Route 53 può anche essere configurato con record di failover che utilizzano i controlli dell'integrità per rispondere alle query con soli indirizzi IP integri. Per tutti i record DNS utilizzati per il failover, specifica un valore TTL (time to live) breve (ad esempio, 60 secondi o meno) per evitare che la memorizzazione nella cache dei record impedisca il ripristino (i record alias di Route 53 forniscono i TTL appropriati).

Passaggi aggiuntivi quando si utilizzano più Regioni AWS

1. Replica tutto il sistema operativo (OS) e il codice dell'applicazione utilizzati dal carico di lavoro nelle Regioni selezionate. Se necessario, replica le Amazon Machine Image (AMI) utilizzate dalle istanze EC2 utilizzando soluzioni come Amazon EC2 Image Builder. Replica le immagini di container archiviate nei registri utilizzando soluzioni come la replica tra Regioni di Amazon ECR. Abilita la replica regionale per tutti i bucket Amazon S3 utilizzati per archiviare le risorse dell'applicazione.
2. Distribuisci le risorse di elaborazione e i metadati di configurazione (come i parametri archiviati in AWS Systems Manager Parameter Store) in più Regioni. Utilizza le stesse procedure descritte nei passaggi precedenti, ma replica la configurazione per ogni Regione utilizzata per il carico di lavoro. Utilizza soluzioni infrastructure as code, ad esempio AWS CloudFormation, per riprodurre in modo uniforme le configurazioni tra le Regioni. Se utilizzi una Regione secondaria in una configurazione fiamma pilota per il disaster recovery, puoi ridurre il numero di risorse di elaborazione a un valore minimo per risparmiare sui costi, con un corrispondente aumento del tempo di ripristino.
3. Replica i dati dalla Regione primaria alle Regioni secondarie.
   a. Le tabelle globali di Amazon DynamoDB forniscono repliche globali dei dati in cui è possibile scrivere da qualsiasi Regione supportata. Con altri servizi dati gestiti da AWS, come Amazon RDS, Amazon Aurora e Amazon Elasticache, si designano una Regione primaria (lettura/scrittura) e Regioni di replica (sola lettura). Per informazioni dettagliate sulla replica regionale, consulta le guide per l'utente e per gli sviluppatori dei rispettivi servizi.
   b. Se esegui un database autogestito, organizza la replica in più Regioni in base alle istruzioni o alle best practice dell'applicazione. Informati sulle procedure di failover della tua applicazione.

   c. Se il carico di lavoro utilizza AWS EventBridge, potrebbe essere necessario inoltrare eventi selezionati dalla Regione primaria alle Regioni secondarie. A tal fine, specifica i bus di eventi nelle Regioni secondarie come destinazioni per gli eventi corrispondenti nella Regione primaria.

4. Considera se e in che misura usare chiavi di crittografia identiche tra le varie Regioni. Un approccio tipico, che consente di bilanciare sicurezza e facilità d'uso, consiste nell'utilizzare chiavi con ambito di Regione per i dati e l'autenticazione a livello di Regione e utilizzare chiavi con ambito globale per la crittografia dei dati replicati tra le diverse Regioni. AWS Key Management Service (KMS) supporta chiavi multiregionali per distribuire e proteggere in modo sicuro le chiavi condivise tra le Regioni.
5. Prendi in considerazione AWS Global Accelerator per migliorare la disponibilità dell'applicazione indirizzando il traffico verso Regioni che contengono endpoint integri.

Risorse

Best practice correlate:

- REL02-BP03 Verifica che l'allocazione delle sottoreti IP consenta l'espansione e la disponibilità
- REL11-BP05 Utilizzo della stabilità statica per evitare un comportamento bimodale
- REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati

Documenti correlati:

- Infrastruttura globale di AWS
- White paper: AWS Fault Isolation Boundaries
- Resilience in Amazon EC2 Auto Scaling
- Amazon EC2 Auto Scaling: Example: Distribute instances across Availability Zones
- How EC2 Image Builder works
- How Amazon ECS places tasks on container instances (includes Fargate)
- Resilienza in AWS Lambda
- Amazon S3: Replicating objects overview
- Private image replication in Amazon ECR
- Global Tables: Multi-Region Replication with DynamoDB
- Amazon Elasticache for Redis OSS: Replication across Regioni AWS using global datastores

- Resilience in Amazon RDS
- Using Amazon Aurora global databases
- AWS Global Accelerator Developer Guide
- Multi-Region keys in AWS KMS
- Amazon Route 53: Configuring DNS failover
- Amazon Application Recovery Controller (ARC) Developer Guide
- Sending and receiving Amazon EventBridge events between Regioni AWS
- Creating a Multi-Region Application with AWS Services blog series
- Disaster Recovery (DR) Architecture on AWS, Part I: Strategies for Recovery in the Cloud
- Disaster Recovery (DR) Architecture on AWS, Part III: Pilot Light and Warm Standby

Video correlati:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications
- AWS re:Invent 2019: Innovation and operation of the AWS global network infrastructure

REL10-BP02 Ripristino automatico dei componenti vincolati a una singola posizione

Se i componenti del carico di lavoro possono essere eseguiti in una sola zona di disponibilità o in un data center on-premises, devi rendere possibile la ricostruzione completa del carico di lavoro in base agli obiettivi di ripristino definiti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Se, a causa di vincoli tecnologici, non è possibile seguire le linee guida per distribuire il carico di lavoro in più posizioni, è necessario implementare un percorso alternativo mirato alla resilienza. È necessario automatizzare la possibilità di ricreare l'infrastruttura necessaria, ridistribuire le applicazioni e ricreare i dati necessari per questi casi.

Ad esempio, Amazon EMR lancia tutti i nodi per un determinato cluster nella stessa zona di disponibilità perché l'esecuzione di un cluster nella stessa zona migliora le prestazioni dei flussi di lavoro poiché fornisce una velocità di accesso ai dati più elevata. Se questo componente è necessario per la resilienza del carico di lavoro, è necessario disporre di un modo per implementare nuovamente il cluster e i relativi dati. Inoltre, per Amazon EMR, è necessario effettuare il provisioning

della ridondanza in modi diversi dall'utilizzo di Multi-AZ. Puoi effettuare il provisioning di più nodi. Utilizzando il file system EMR (EMRFS), i dati in EMR possono essere memorizzati in Amazon S3, che a sua volta può essere replicato su più zone di disponibilità o Regioni AWS.

Analogamente, Amazon Redshift per impostazione predefinita effettua il provisioning del cluster in una zona di disponibilità casuale all'interno della Regione AWS selezionata. Viene effettuato il provisioning di tutti i nodi del cluster nella stessa zona.

Per carichi di lavoro basati su server stateful implementati in un data center on-premises, puoi usare AWS Elastic Disaster Recovery per proteggerli in AWS. Se il carico di lavoro è già ospitato in AWS, Elastic Disaster Recovery ti consente di proteggerlo in una zona di disponibilità o regione alternativa. Elastic Disaster Recovery sfrutta la replica a livello di blocco continua in un'area di gestione temporanea leggera per fornire il ripristino rapido e affidabile di applicazioni on-premises e basate sul cloud.

Passaggi dell'implementazione

1. Implementa l'autoriparazione. Implementa istanze o container utilizzando, quando possibile, il dimensionamento automatico. Se non è possibile utilizzare il dimensionamento automatico, utilizza il ripristino automatico per istanze EC2 o implementa l'automazione di autoriparazione in base agli eventi del ciclo di vita di container Amazon EC2 o ECS.
   - Utilizza gruppi Amazon EC2 Auto Scaling per carichi di lavoro di container e istanze che non richiedono un indirizzo IP di una singola istanza, un indirizzo IP privato, un indirizzo IP elastico o metadati di istanza.
     - È possibile usare i dati utente del modello di avvio per implementare l'automazione per la riparazione automatica della maggior parte dei carichi di lavoro.
   - Utilizza il ripristino delle istanze Amazon EC2 per carichi di lavoro che richiedono un indirizzo ID di una singola istanza, un indirizzo IP privato, un indirizzo IP elastico e metadati di istanza.
     - Il ripristino automatico invierà avvisi sullo stato del ripristino a un argomento SNS quando viene rilevato l'errore dell'istanza.
   - Utilizza gli eventi del ciclo di vita di istanze Amazon EC2 o gli eventi Amazon ECS per automatizzare l'autoriparazione dove non è possibile utilizzare il dimensionamento automatico o il ripristino EC2.
     - Utilizza gli eventi per richiamare l'automazione che riparerà il tuo componente secondo la logica di processo richiesta.
   - Utilizza AWS Elastic Disaster Recovery per proteggere i carichi di lavoro stateful limitati a una singola posizione.

Risorse

Documenti correlati:

- Amazon ECS events
- Amazon EC2 Auto Scaling lifecycle hooks
- Recupero di un'istanza
- Ridimensionamento automatico del servizio
- What Is Amazon EC2 Auto Scaling?
- AWS Elastic Disaster Recovery

REL10-BP03 Utilizzo di architetture a scomparti per limitare la portata dell'impatto

Implementa architetture a scomparti (note anche come architetture basate su celle) per limitare l'effetto di un guasto all'interno di un carico di lavoro a un numero ridotto di componenti.

Risultato desiderato: un'architettura basata su celle utilizza più istanze isolate di un carico di lavoro, ciascuna delle quali è nota come cella. Ogni cella è indipendente, non condivide lo stato con altre celle e gestisce un sottoinsieme delle richieste complessive del carico di lavoro. Questo approccio riduce il possibile impatto di un errore, ad esempio un aggiornamento software non valido, a una singola cella e alle richieste elaborate. Se un carico di lavoro usa 10 celle per gestire 100 richieste e si verifica un errore, il 90% delle richieste complessive non sarà interessato dall'errore.

Anti-pattern comuni:

- Aumento illimitato delle celle.
- Applicazione di aggiornamenti o implementazioni del codice in tutte le celle contemporaneamente.
- Condivisione dello stato dei componenti tra celle (con l'eccezione del livello di instradamento).
- Aggiunta di logica di business o instradamento complessa al livello di instradamento.
- Le interazioni tra celle non sono ridotte al minimo.

Vantaggi dell'adozione di questa best practice: limitazione alla cella stessa di molti tipi comuni di errori, a garanzia di un ulteriore isolamento dei guasti, grazie alle architetture basate su celle. Questi limiti relativi agli errori possono garantire resilienza in caso di determinati tipi di errori, altrimenti difficili da contenere, come implementazioni di codice non riuscite o richieste danneggiate o che richiamano una modalità di errore specifica (nota anche come richieste poison pill).

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Su una nave gli scomparti permettono di limitare la falla di uno scafo a una sola sezione dello scafo. In sistemi complessi, questo modello viene spesso replicato per consentire l'isolamento degli errori. Le limitazioni per l'isolamento degli errori riducono l'effetto di un errore all'interno di un carico di lavoro a un numero limitato di componenti. I componenti al di fuori della barriera non subiscono gli effetti del guasto. Utilizzando più barriere per l'isolamento dei guasti, puoi limitare l'impatto sul carico di lavoro. In AWS i clienti possono usare più zone di disponibilità e regioni per fornire l'isolamento degli errori, ma questo concetto può essere esteso anche all'architettura del carico di lavoro.

Il carico di lavoro complessivo viene partizionato in celle tramite una chiave di partizione. Questa chiave deve essere allineata alla granularità del servizio o al modo naturale in cui il carico di lavoro del servizio può essere suddiviso con interazioni minime tra celle. Esempi di chiavi di partizione sono un ID cliente, un ID risorsa o qualsiasi altro parametro facilmente accessibile nella maggior parte delle chiamate API. Un livello di instradamento alle celle distribuisce le richieste a singole celle in base alla chiave di partizione e presenta un unico endpoint ai client.

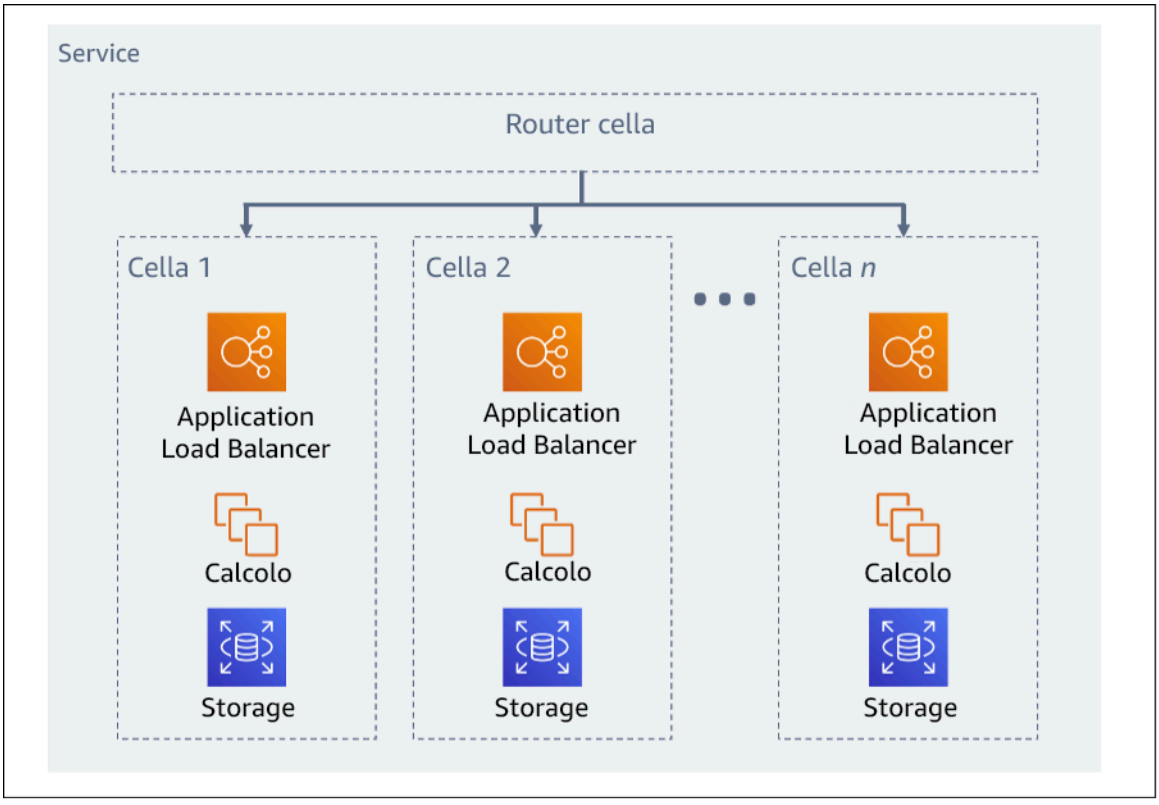


Figura 11: architettura basata su celle

Passaggi dell'implementazione

Nel progettare un'architettura basata su celle, devi tenere conto di diversi aspetti della progettazione:

1. Chiave di partizione: presta particolare attenzione alla scelta della chiave di partizione.
   - Questa deve essere allineata alla granularità del servizio o al modo naturale in cui il carico di lavoro del servizio può essere suddiviso con interazioni minime tra celle. Alcuni esempi sono `customer ID` o `resource ID`.
   - La chiave di partizione deve essere disponibile in tutte le richieste, direttamente o in modo da poter essere facilmente dedotta in modo deterministico da altri parametri.
2. Mappatura persistente delle celle: i servizi a monte devono interagire solo con una singola cella per l'intero ciclo di vita delle risorse correlate.
   - A seconda del carico di lavoro, può essere necessaria una strategia di migrazione delle celle per la migrazione dei dati da una cella a un'altra. Un possibile scenario in cui è necessaria la migrazione delle celle è quando una risorsa o un utente specifico nel carico di lavoro diventa troppo grande e richiede una cella dedicata.
   - Le celle non devono condividere lo stato o i componenti.
   - Di conseguenza, l'interazione tra celle deve essere evitata o mantenuta al minimo, in quanto le interazioni creano dipendenze tra le celle e riducono quindi i vantaggi forniti dall'isolamento degli errori.
3. Livello di instradamento: il livello di instradamento è un componente condiviso tra celle, pertanto non può basarsi sulla stessa strategia di compartimentazione delle celle.
   - È consigliabile che il livello di instradamento distribuisca richieste a singole celle usando un algoritmo di mappatura delle partizioni efficiente in termini di risorse di calcolo, ad esempio combinando funzioni hash crittografiche e aritmetica modulare per mappare le chiavi di partizione alle celle.
   - Per evitare l'impatto su più celle, il livello di instradamento deve restare il più semplice e orizzontalmente scalabile possibile, evitando logica di business complessa in questo livello. Questo approccio offre il vantaggio aggiuntivo di semplificare la comprensione del suo comportamento previsto in ogni momento, permettendo test esaustivi. Come illustrato da Colm MacCárthaigh in Reliability, constant work, and a good cup of coffee, progettazioni semplici e schemi di lavoro costanti si traducono in sistemi affidabili e nella riduzione dell'antifragilità.
4. Dimensione delle celle: le celle devono avere una dimensione massima che non deve essere superata
   - La dimensione massima va identificata attraverso l'esecuzione di test completi, fino a raggiungere i punti di rottura e definire i margini operativi. Per ulteriori informazioni su come implementare procedure di test, consulta REL07-BP04 Load Testa il tuo carico di lavoro.

- L'aumento del carico di lavoro complessivo deve essere gestito tramite l'aggiunta di celle, in modo da poterlo dimensionare in base al crescere della domanda.

5. Strategie multi-AZ o multi-regione: si consiglia di utilizzare più livelli di resilienza per proteggersi da diversi domini di errore.
   - Per la resilienza, devi utilizzare un approccio che costruisca livelli di difesa. Un livello protegge dalle interruzioni minime e più comuni attraverso la creazione di un'architettura a disponibilità elevata tramite più zone di disponibilità. Un altro livello di difesa è destinato a proteggere da eventi rari come disastri naturali diffusi e interruzioni a livello regionale. Questo secondo livello implica l'architettura dell'applicazione in modo che si estenda su più Regioni AWS. L'implementazione di una strategia multi-regione per il tuo carico di lavoro aiuta a proteggerlo da disastri naturali diffusi che colpiscono un'ampia regione geografica di un paese o da guasti tecnici di portata regionale. Tieni presente che l'implementazione di un'architettura multi-regione può essere molto complessa e di solito non è necessaria per la maggior parte dei carichi di lavoro. Per ulteriori dettagli, consulta REL10-BP01 Implementazione del carico di lavoro in diversi luoghi.
6. Implementazione del codice: è preferibile una strategia di implementazione del codice scaglionata rispetto all'implementazione simultanea di modifiche al codice in tutte le celle.
   - In questo modo, è possibile ridurre al minimo eventuali errori in più celle a causa di un'implementazione non corretta o dell'errore umano. Per ulteriori informazioni, consulta Automatizzazione di distribuzioni pratiche e sicure.

Risorse

Best practice correlate:

- REL07-BP04 Load Testa il tuo carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi

Documenti correlati:

- Reliability, constant work, and a good cup of coffee
- AWS and Compartmentalization
- Isolamento del carico di lavoro utilizzando lo sharding casuale
- Automatizzazione di distribuzioni pratiche e sicure

Video correlati:

- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small
- AWS re:Invent 2018: How AWS Minimizes the Blast Radius of Failures (ARC338)
- Shuffle-sharding: AWS re:Invent 2019: Introducing The Amazon Builders' Library (DOP328)
- AWS Summit ANZ 2021 - Everything fails, all the time: Designing for resilience

REL 11. Come si progetta il carico di lavoro affinché resista ai guasti dei componenti?

I carichi di lavoro con requisiti di disponibilità elevata e MTTR (Mean Time To Recovery) basso devono essere progettati per garantire la resilienza.

Best practice

- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP02 Failover e passaggio a risorse integre
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL11-BP04 Fare affidamento al piano dati invece che al piano di controllo (control-plane) durante il ripristino
- REL11-BP05 Utilizzo della stabilità statica per evitare un comportamento bimodale
- REL11-BP06 Invio di notifiche quando gli eventi influiscono sulla disponibilità
- REL11-BP07 Progettazione del prodotto in modo da soddisfare gli obiettivi di disponibilità e gli accordi sul livello di servizio (SLA) per i tempi di attività

REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

Monitora costantemente lo stato del carico di lavoro, in modo che tu e i tuoi sistemi automatizzati siate consapevoli di errori o guasti non appena si verificano. Monitora gli indicatori chiave di prestazioni (KPI) in base al valore aziendale.

Tutti i meccanismi di ripristino e correzione devono essere in grado di rilevare rapidamente i problemi. I guasti tecnici devono essere rilevati prima in modo che possano essere risolti. Tuttavia, la disponibilità si basa sulla capacità del carico di lavoro di fornire valore aziendale, quindi gli indicatori chiave di prestazione (KPI) che misurano questo aspetto devono far parte della strategia di rilevamento e correzione.

Risultato desiderato: i componenti essenziali di un carico di lavoro vengono monitorati in modo indipendente per rilevare guasti e fornire avvisi quando e dove si verificano.

Anti-pattern comuni:

- Non sono stati configurati allarmi, pertanto le interruzioni si verificano senza notifica.
- Gli allarmi esistono, ma a soglie che non forniscono tempo adeguato per reagire.
- I parametri non vengono raccolti abbastanza spesso da soddisfare l'obiettivo del tempo di ripristino (RTO)
- Solo le interfacce del carico di lavoro rivolte al cliente vengono monitorate attivamente.
- Viene effettuata solo la raccolta di parametri tecnici, senza includere quelli delle funzioni aziendali.
- Non è presente alcun parametro che misuri l'esperienza utente del carico di lavoro.
- Vengono creati troppi monitoraggi.

Vantaggi dell'adozione di questa best practice: eseguire un monitoraggio appropriato a tutti i livelli consente di ridurre i tempi di rilevamento, velocizzando quindi il ripristino.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Identifica tutti i carichi di lavoro che verranno esaminati per il monitoraggio. Dopo aver identificato tutti i componenti del carico di lavoro da monitorare, devi determinare l'intervallo di monitoraggio. L'intervallo di monitoraggio ha un impatto diretto sulla velocità con cui il ripristino viene avviato, che dipende dal tempo impiegato per rilevare un errore. Il tempo medio di rilevamento (MTTD) è il tempo che intercorre tra il verificarsi di un guasto e l'inizio delle operazioni di riparazione. L'elenco dei servizi deve essere ampio e completo.

Il monitoraggio deve includere tutti i livelli dello stack applicativo, come applicazione, piattaforma, infrastruttura e rete.

La strategia di monitoraggio deve tenere in considerazione l'impatto dei guasti nell'area grigia. Per ulteriori informazioni sui guasti nell'area grigia, consulta Gray failures nel whitepaper Advanced Multi-AZ Resilience Patterns.

Passaggi dell'implementazione

- L'intervallo di monitoraggio dipende dalla velocità con cui è necessario ripristinare Il tempo di ripristino dipende dal tempo necessario a ripristinare, perciò è necessario determinare la frequenza della raccolta considerando tale tempo e l'obiettivo del tempo di ripristino (RTO)
- Configura il monitoraggio dettagliato per componenti e servizi gestiti.
  - Determina se è necessario un monitoraggio dettagliato per le istanze EC2 e Auto Scaling. Il monitoraggio dettagliato fornisce metriche a intervalli di un minuto, mentre il monitoraggio predefinito fornisce metriche a intervalli di cinque minuti.
  - Determina se è necessario un monitoraggio avanzato per RDS. Il monitoraggio avanzato utilizza un agente sulle istanze RDS per ottenere informazioni utili su diversi processi o thread.
  - Determina i requisiti di monitoraggio dei componenti serverless critici per Lambda, API Gateway, Amazon EKS, Amazon ECS e tutti i tipi di bilanciatori del carico.
  - Determina i requisiti di monitoraggio dei componenti di archiviazione per Amazon S3, Amazon FSx, Amazon EFS e Amazon EBS.
- Crea parametri personalizzati per misurare indicatori chiave di prestazione (KPI) aziendali. I carichi di lavoro implementano funzioni aziendali fondamentali, che devono essere utilizzate come KPI che aiutano a identificare quando si verifica un problema indiretto.
- Monitora la presenza di errori nell'esperienza utente tramite le canary degli utenti Il test sintetico delle transazioni (noto anche come "test canary", ma da non confondere con le distribuzioni canary) in grado di eseguire e simulare il comportamento dei clienti è uno dei processi di test più importanti. Esegui questi test costantemente sugli endpoint del carico di lavoro da diverse posizioni remote.
- Crea parametri personalizzati che monitorino l'esperienza dell'utente. Dotare l'esperienza del cliente di strumenti consente di determinare quando essa peggiora.
- Imposta gli allarmi per rilevare quando una qualsiasi parte del carico di lavoro non funziona correttamente e per indicare quando effettuare il dimensionamento automatico delle risorse. È possibile mostrare visivamente gli avvisi sui pannelli di controllo, inviarli tramite Amazon SNS o e-mail e utilizzarli con Auto Scaling per aumentare o ridurre le risorse del carico di lavoro.
- Crea pannelli di controllo per visualizzare i parametri. Utilizza i pannelli di controllo per visualizzare tendenze, valori anomali e altri indicatori di potenziali problemi, oppure per fornire un'indicazione dei problemi che potresti voler approfondire.
- Crea il monitoraggio del tracciamento distribuito per i tuoi servizi. Con il monitoraggio distribuito puoi comprendere le prestazioni della tua applicazione e dei relativi servizi sottostanti per identificare e risolvere la causa ultima di problemi ed errori riguardanti le prestazioni.

- Crea sistemi di monitoraggio (utilizzando CloudWatch o X-Ray), pannelli di controllo e raccolta dati in una regione e in un account separati.
- Con AWS Health si ricevono le informazioni sul degrado delle prestazioni del servizio. Si creano notifiche di eventi AWS Health personalizzati per i canali e-mail e chat con Notifiche all'utente AWS e si usano gli strumenti di monitoraggio e avviso con Amazon EventBridge per l'integrazione a livello di codice.

Risorse

Best practice correlate:

- Definizione di disponibilità
- REL11-BP06 Invio di notifiche quando gli eventi influiscono sulla disponibilità

Documenti correlati:

- Amazon CloudWatch Synthetics consente di creare canary dell'utente
- Abilitare o disabilitare il monitoraggio dettagliato della propria istanza
- Monitoraggio avanzato
- Monitoring Your Auto Scaling Groups and Instances Using Amazon CloudWatch
- Publishing Custom Metrics
- Using Amazon CloudWatch Alarms
- Using CloudWatch Dashboards
- Using Cross Region Cross Account CloudWatch Dashboards
- Uso del tracciamento X-Ray tra più regioni e account
- Understanding availability

Video correlati:

- Mitigating gray failures

Esempi correlati:

- One Observability Workshop: Explore X-Ray

Strumenti correlati:

- CloudWatch
- CloudWatch X-Ray

REL11-BP02 Failover e passaggio a risorse integre

Se si verifica un errore in una risorsa, le risorse integre dovrebbero continuare a soddisfare le richieste. Per posizioni compromesse (ad esempio, una zona di disponibilità o una Regione AWS), assicurati di disporre di sistemi che possano eseguire il failover e passare a risorse integre in posizioni non danneggiate.

Durante la progettazione di un servizio, distribuisci il carico tra risorse, zone di disponibilità o regioni. In questo modo, il guasto o la compromissione di una singola risorsa può essere mitigato spostando il traffico sulle risorse integre rimanenti. Considera come vengono rilevati e indirizzati i servizi in caso di guasto.

Progetta i tuoi servizi tenendo a mente il recupero dai guasti. In AWS, progettiamo servizi per ridurre al minimo i tempi di recupero da guasti e l'impatto sui dati. I nostri servizi utilizzano principalmente archivi di dati che riconoscono le richieste solo dopo che queste sono state archiviate in modo duraturo su più repliche in una regione. Sono costruiti con il criterio dell'isolamento basato sulle celle ed utilizzano l'isolamento dei guasti fornito dalle zone di disponibilità. Facciamo ampio uso dell'automazione nelle nostre procedure operative. Ottimizziamo anche la nostra funzionalità di sostituzione e riavvio per un ripristino rapidamente dalle interruzioni.

I modelli e i progetti che consentono il failover variano a seconda dei servizi della AWS. Molti servizi AWS gestiti nativi si trovano in più zone di disponibilità (come Lambda o API Gateway) in modo nativo. Altri servizi AWS (come EC2 ed EKS) richiedono procedure ottimali specifiche per supportare il failover delle risorse o l'archiviazione di dati tra le zone di disponibilità.

Il monitoraggio deve essere impostato per verificare che la risorsa di failover sia integra, tenere traccia dell'avanzamento del failover delle risorse e monitorare il ripristino dei processi aziendali.

Risultato desiderato: i sistemi sono in grado di utilizzare automaticamente o manualmente nuove risorse per il ripristino dopo un evento di deterioramento.

Anti-pattern comuni:

- La pianificazione degli errori non fa parte della fase di pianificazione e progettazione.
- L'obiettivo del tempo di ripristino (RTO) e l'obiettivo del punto di ripristino (RPO) non sono stabiliti.

- Monitoraggio insufficiente per rilevare risorse difettose.
- Isolamento adeguato dei domini di errore.
- Il failover multi-regione non è considerato.
- Il rilevamento dei guasti è troppo sensibile o aggressivo quando si decide di eseguire il failover.
- Non è possibile testare o convalidare il progetto di failover.
- Esecuzione dell'automazione del risanamento automatico, ma senza la notifica della necessità di una correzione.
- Mancanza di un periodo di mitigazione per evitare che l'errore si ripresenti troppo presto.

Vantaggi dell'adozione di questa best practice: è possibile creare sistemi più resilienti che garantiscano l'affidabilità in caso di guasti eseguendo prima un deterioramento lento e poi un ripristino rapido.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I servizi AWS, come Elastic Load Balancing e Amazon EC2 Auto Scaling, consentono di distribuire il carico tra risorse e zone di disponibilità. In questo modo, il guasto di una singola risorsa (come un'istanza EC2) o la compromissione di una zona di disponibilità possono essere mitigati spostando il traffico sulle risorse integre rimanenti.

Per i carichi di lavoro multi-regione, i progetti sono più complicati. Ad esempio, le repliche di lettura multi-regione consentono di implementare i dati su Regioni AWS multiple. Tuttavia, il failover è ancora necessario per promuovere la replica di lettura a principale e quindi indirizzare il traffico verso il nuovo endpoint. Amazon Route 53, Sistema di controllo Amazon per il rispristino di applicazioni (ARC), Amazon CloudFront e AWS Global Accelerator consentono di instradare il traffico nelle Regioni AWS.

I servizi AWS, come Amazon S3, Lambda, API Gateway, Amazon SQS, Amazon SNS, Amazon SES, Amazon Pinpoint, Amazon ECR, AWS Certificate Manager, EventBridge o Amazon DynamoDB sono implementati in automatico in più zone di disponibilità da AWS. In caso di guasto, questi servizi AWS instradano automaticamente il traffico verso posizioni integre. I dati sono archiviati in modo ridondante in più zone di disponibilità e rimangono disponibili.

Per Amazon RDS, Amazon Aurora, Amazon Redshift, Amazon EKS o Amazon ECS, una delle opzioni di configurazione è Multi-AZ. AWS può indirizzare il traffico verso l'istanza integra in caso di

avvio del failover. Questa azione di failover può essere intrapresa direttamente da AWS o su richiesta del cliente.

Per le istanze Amazon EC2, Amazon Redshift, le attività Amazon ECS o i pod Amazon EKS, sei tu a scegliere le zone di disponibilità in cui effettuare l'implementazione. Per alcuni progetti, Elastic Load Balancing fornisce la soluzione per rilevare le istanze in zone non integre e instradare il traffico verso quelle integre. Elastic Load Balancing può inoltre instradare il traffico verso componenti nel tuo data center on-premises.

Per il failover del traffico multi-regione, il reindirizzamento può sfruttare Amazon Route 53, Sistema di controllo Amazon per il ripristino di applicazioni, AWS Global Accelerator, Route 53, DNS privato per VPC o CloudFront per fornire una modalità di definizione dei domini Internet e assegnare policy di instradamento, compresi i controlli dell'integrità, per instradare il traffico verso regioni integre. AWS Global Accelerator fornisce indirizzi IP statici che operano come punto di ingresso fisso all'applicazione, che indirizzano il traffico verso gli endpoint delle Regioni AWS di propria scelta utilizzando la rete AWS globale anziché Internet per prestazioni e affidabilità migliori.

Passaggi dell’implementazione

- Crea progetti di failover per tutte le applicazioni e i servizi appropriati. Isola ogni componente dell’architettura e crea progetti di failover che soddisfino l’RTO e l’RPO per ogni componente.
- Configura ambienti inferiori (come sviluppo o test) con tutti i servizi necessari per disporre di un piano di failover. Implementa le soluzioni utilizzando il modello infrastructure as code (IaC) per garantire la ripetibilità.
- Configura un sito di ripristino, ad esempio una seconda regione, per implementare e testare i progetti di failover. Se necessario, le risorse per i test possono essere configurate temporaneamente per limitare i costi aggiuntivi.
- Determina quali piani di failover sono automatizzati da AWS, quali possono essere automatizzati da un processo DevOps e quali possono essere manuali. Documenta e misura l’RTO e l’RPO di ogni servizio.
- Crea un playbook per il failover e includi tutti i passaggi necessari per eseguire il failover di ogni risorsa, applicazione e servizio.
- Crea un playbook di failback e includi tutti i passaggi per eseguire il failback (con tempistiche) di ogni risorsa, applicazione e servizio.
- Crea un piano per avviare e testare il playbook. Usa simulazioni e test del caos per testare i passaggi e l’automazione del playbook.

- Per posizioni compromesse (ad esempio una zona di disponibilità o una Regione AWS), assicurati di disporre di sistemi che possano eseguire il failover e passare a risorse integre in posizioni non danneggiate. Verifica la quota, i livelli di dimensionamento automatico e le risorse in esecuzione prima dei test di failover.

Risorse

Best practice Well-Architected correlate:

- REL13- Pianificazione per il disaster recovery (DR
- REL10 - Utilizzo dell'isolamento dei guasti per proteggere il carico di lavoro

Documenti correlati:

- Impostazione di obiettivi RTO e RPO
- Failover utilizzando il routing ponderato Route 53
- Disaster Recovery with Amazon Application Recovery Controller
- EC2 with autoscaling
- EC2 Deployments - Multi-AZ
- ECS Deployments - Multi-AZ
- Switch traffic using Amazon Application Recovery Controller
- Lambda with an Application Load Balancer and Failover
- ACM Replication and Failover
- Parameter Store Replication and Failover
- ECR cross region replication and Failover
- Secrets manager cross region replication configuration
- Enable cross region replication for EFS and Failover
- EFS Cross Region Replication and Failover
- Networking Failover
- S3 Endpoint failover using MRAP
- Crea una replica tra regioni per S3
- Guidance for Cross Region Failover and Graceful Failback on AWS
- Failover using multi-region global accelerator

- Failover with DRS

Esempi correlati:

- Disaster Recovery on AWS
- Elastic Disaster Recovery on AWS

REL11-BP03 Automatizzazione della riparazione a tutti i livelli

Al rilevamento di un guasto, utilizza funzionalità automatizzate per eseguire azioni da correggere. I guasti possono essere riparati automaticamente tramite meccanismi di servizio interni oppure riavviando o rimuovendo le risorse tramite azioni correttive.

Per applicazioni gestite dal cliente e per il ripristino tra regioni, è possibile attingere a modelli di ripristino e processi di riparazione automatizzati dalle best practice esistenti.

La possibilità di riavviare o rimuovere una risorsa è uno strumento importante per risolvere i guasti. Una best practice consiste nel rendere i servizi stateless, ove possibile. In questo modo si evita la perdita di dati o di disponibilità durante il riavvio della risorsa. Nel cloud è possibile, e in genere si dovrebbe, sostituire l'intera risorsa (ad esempio, un'istanza di calcolo o una funzione serverless) come parte del riavvio. Il riavvio stesso è un modo semplice e affidabile per eseguire il ripristino in caso di guasto. Molti tipi diversi di guasto si verificano nei carichi di lavoro. Possono verificarsi guasti a livello di hardware, software, comunicazione e operazioni.

Il riavvio o i nuovi tentativi come pratiche risolutive si applicano anche alle richieste di rete. Adotta lo stesso approccio di ripristino sia a un timeout di rete sia a un guasto di dipendenza in cui la dipendenza restituisce un guasto. Entrambi gli eventi hanno un effetto simile sul sistema, quindi piuttosto che tentare di trasformare entrambi gli eventi in un caso speciale, adotta una strategia analoga di nuovi tentativi limitati con un jitter e un backoff esponenziali. La capacità di riavvio è un meccanismo di ripristino presente nelle architetture di cluster ROC (Recovery-oriented computing) e ad alta disponibilità.

Risultato desiderato: vengono eseguite azioni automatiche di risoluzione a seguito del rilevamento di un errore.

Anti-pattern comuni:

- Provisioning di risorse senza dimensionamento automatico.
- Implementazione individuale di applicazioni in istanze/container.

- Implementazione di applicazioni che non possono essere distribuite in più posizioni senza utilizzare il ripristino automatico.
- Riparazione manuale delle applicazioni che il dimensionamento e il ripristino automatici non sono stati in grado di riparare.
- Nessuna automazione dei database di failover.
- Mancanza di metodi automatizzati per reinstradare il traffico verso nuovi endpoint.
- Nessuna replica dell'archiviazione.

Vantaggi dell'adozione di questa best practice: la riparazione automatica può ridurre il tempo medio di ripristino e migliorare la disponibilità.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I progetti per Amazon EKS o altri servizi Kubernetes devono includere il numero minimo e massimo di repliche o di stateful set e la dimensione minima dei cluster e dei gruppi di nodi. Questi meccanismi forniscono una quantità minima di risorse di elaborazione continuamente disponibili mentre riparano automaticamente eventuali guasti utilizzando il piano di controllo (control-plane) Kubernetes.

I modelli di progettazione a cui si accede tramite un bilanciatore del carico che utilizza cluster di calcolo dovrebbero sfruttare i gruppi Auto Scaling. Elastic Load Balancing (ELB) distribuisce automaticamente il traffico delle applicazioni in entrata su più destinazioni e applicazioni virtuali in una o più zone di disponibilità (AZ).

I progetti basati su cluster computing che non utilizzano il bilanciamento del carico devono avere dimensioni progettate per la perdita di almeno un nodo. Ciò consentirà al servizio di rimanere in esecuzione con una capacità potenzialmente ridotta durante il ripristino di un nuovo nodo. Ecco alcuni esempi di servizi: Mongo, DynamoDB Accelerator, Amazon Redshift, Amazon EMR, Cassandra, Kafka, MSK-EC2, Couchbase, ELK e il Servizio OpenSearch di Amazon Molti di questi servizi possono essere progettati con funzionalità di riparazione automatica aggiuntive. Alcune tecnologie di cluster devono generare un avviso in caso di perdita di un nodo attivando un flusso di lavoro automatico o manuale per creare un nuovo nodo. È possibile automatizzare questo flusso di lavoro utilizzando AWS Systems Manager per risolvere rapidamente i problemi.

Amazon EventBridge può essere utilizzato per monitorare e filtrare eventi come allarmi CloudWatch o modifiche di stato in altri servizi AWS. In base alle informazioni sugli eventi, può quindi richiamare

AWS Lambda, Systems Manager Automation o altre destinazioni per eseguire una logica di riparazione personalizzata sul tuo carico di lavoro. Amazon EC2 Auto Scaling può essere configurato per verificare lo stato dell'istanza EC2. Se l'istanza è in uno stato diverso da quello in esecuzione o se lo stato del sistema è danneggiato, Amazon EC2 Auto Scaling considera l'istanza come non integra e avvia un'istanza sostitutiva. Per le sostituzioni su larga scala (ad esempio la perdita di un'intera zona di disponibilità), è preferibile adottare la stabilità statica per ottenere un'elevata disponibilità.

Passaggi dell'implementazione

- Utilizza i gruppi Auto Scaling per implementare livelli in un carico di lavoro. Auto Scaling è in grado di eseguire il risanamento automatico sulle applicazioni stateless e aggiungere o rimuovere capacità.
- Per le istanze di calcolo menzionate in precedenza, utilizza il bilanciamento del carico e scegli il tipo di bilanciatore del carico adeguato.
- Prendi in considerazione la riparazione per Amazon RDS Utilizzando le istanze di standby, puoi configurare il failover automatico sulle stesse. Per le repliche in lettura Amazon RDS, è necessario un flusso di lavoro automatizzato per rendere primaria una replica di lettura.
- Implementa il ripristino automatico sulle istanze EC2 che includono applicazioni distribuite non implementabili in più posizioni e possono tollerare il riavvio in caso di guasti. Il ripristino automatico può essere utilizzato per sostituire l'hardware guasto e riavviare l'istanza quando l'applicazione non è in grado di essere distribuita in più posizioni. Vengono conservati i metadati dell'istanza e gli indirizzi IP associati, nonché i volumi EBS e i punti di montaggio su Amazon Elastic File System o file system per Lustre e Windows. Grazie a AWS OpsWorks, puoi configurare la riparazione automatica delle istanze EC2 a livello del layer.
- Implementa il ripristino automatico utilizzando AWS Step Functions e AWS Lambda quando non è possibile utilizzare il dimensionamento automatico o il ripristino automatico oppure quando il ripristino automatico non riesce. Quando non puoi utilizzare il dimensionamento automatico né il ripristino automatico o il ripristino automatico non riesce, puoi automatizzare la riparazione utilizzando AWS Step Functions e AWS Lambda.
- Amazon EventBridge può essere utilizzato per monitorare e filtrare eventi come allarmi CloudWatch o modifiche di stato in altri servizi AWS. In base alle informazioni sugli eventi, può quindi richiamare AWS Lambda (o altre destinazioni) per eseguire una logica di riparazione personalizzata sul tuo carico di lavoro.

Risorse

Best practice correlate:

- Definizione di disponibilità
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

Documenti correlati:

- Funzionamento di AWS Auto Scaling
- Ripristino automatico di Amazon EC2
- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- What is Amazon FSx for Lustre?
- What is Amazon FSx for Windows File Server?
- AWS OpsWorks: utilizzo della riparazione automatica per sostituire le istanze con errore
- Cos'è AWS Step Functions?
- Cos'è AWS Lambda?
- What Is Amazon EventBridge?
- Using Amazon CloudWatch Alarms
- Amazon RDS Failover
- SSM - Systems Manager Automation
- Best practice per architetture resilienti

Video correlati:

- Automatically Provision and Scale OpenSearch Service
- Amazon RDS Failover Automatically

Esempi correlati:

- Workshop su Amazon RDS Failover

Strumenti correlati:

- CloudWatch
- CloudWatch X-Ray

REL11-BP04 Fare affidamento al piano dati invece che al piano di controllo (control-plane) durante il ripristino

Il piano di controllo (control-plane) forniscono le API amministrative utilizzate per creare, leggere e descrivere, aggiornare, eliminare ed elencare (CRUDL) risorse, mentre i piani dati gestiscono il traffico quotidiano del servizio. Durante l'implementazione di risposte di ripristino o mitigazione a eventi che possono influire sulla resilienza, concentrati sull'utilizzo di un numero minimo di operazioni del piano di controllo (control-plane) per ripristinare, ridimensionare, ristabilire, riparare il servizio o eseguirne il failover. Le operazioni del piano dati dovrebbero avere la precedenza su qualsiasi attività durante questi eventi che causano deterioramento.

Ad esempio, le seguenti sono tutte azioni del piano di controllo (control-plane): avvio di una nuova istanza di calcolo, creazione di storage a blocchi e descrizione dei servizi di coda. Quando avvii istanze di calcolo, il piano di controllo (control-plane) deve eseguire diverse attività, come trovare un host fisico con capacità, allocare interfacce di rete, preparare volumi di storage a blocchi locali, generare credenziali e aggiungere regole di sicurezza. I piani di controllo (control-plane) tendono ad avere un'orchestrazione complicata.

Risultato desiderato: quando lo stato di risorsa viene compromesso, il sistema è in grado di ripristinarsi automaticamente o manualmente spostando il traffico da risorse danneggiate a risorse integre.

Anti-pattern comuni:

- Dipendenza dalla modifica dei record DNS per reindirizzare il traffico.
- Dipendenza dalle operazioni di dimensionamento del piano di controllo (control-plane) per sostituire i componenti danneggiati a causa di un provisioning delle risorse insufficiente.
- Affidarsi ad azioni intense, multiservizio e multi-API del piano di controllo (control-plane) per porre rimedio a qualsiasi categoria di deterioramento.

Vantaggi dell'adozione di questa best practice una maggiore percentuale di successo in termini di riparazione automatica può ridurre il tempo medio di ripristino e migliorare la disponibilità del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio. Per determinati tipi di degrado del servizio, i piani di controllo (control-plane) sono interessati. Le dipendenze dall'uso intenso del piano di controllo (control-plane) per la riparazione possono aumentare il tempo di ripristino (RTO) e il tempo medio di ripristino (MTTR).

Guida all'implementazione

Per limitare le azioni del piano dati, esegui una valutazione del servizio determinare le azioni necessarie per ripristinarlo.

Sfrutta Amazon Application Recovery Controller per spostare il traffico DNS. Queste funzionalità monitorano continuamente la capacità dell'applicazione di ristabilirsi dai guasti e consentono di controllarne il ripristino su più Regioni AWS, zone di disponibilità e on-premises.

Le policy di instradamento di Route 53 utilizzano il piano di controllo (control-plane), quindi non fare affidamento su di esso per il ripristino. I piani dati di Route 53 rispondono alle query DNS ed eseguono e valutano i controlli dell'integrità. Sono distribuiti a livello globale e progettati per un accordo sul livello di servizio (SLA) con disponibilità pari al 100%.

Le API e le console di gestione di Route 53, dove si creano, aggiornano ed eliminano le risorse di Route 53, funzionano su piani di controllo (control-plane) progettati per privilegiare la forte coerenza e la durata necessarie per la gestione del DNS. A tal fine, i piani di controllo (control-plane) sono situati in un'unica regione: Stati Uniti orientali (Virginia settentrionale). Sebbene entrambi i sistemi siano costruiti per essere molto affidabili, i piani di controllo (control-plane) non sono inclusi nello SLA. Possono verificarsi eventi rari in cui la progettazione resiliente del piano dati consente di mantenere la disponibilità mentre i piani di controllo (control-plane) non lo fanno. Per i meccanismi di disaster recovery e failover, utilizzare le funzioni del piano dati per garantire la migliore affidabilità possibile.

Progetta la tua infrastruttura di elaborazione in modo che sia staticamente stabile per evitare di utilizzare il piano di controllo durante un incidente. Ad esempio, se utilizzi istanze Amazon EC2, evita di effettuare il provisioning manuale di nuove istanze o di chiedere ai gruppi Auto Scaling di aggiungere istanze in risposta. Per ottenere i massimi livelli di resilienza, è necessario fornire una capacità sufficiente nel cluster utilizzato per il failover. Se è necessario limitare questa soglia di capacità, imposta limitazioni (della larghezza di banda della rete) sull'intero sistema end-to-end per limitare in modo sicuro il traffico totale che raggiunge il set limitato di risorse.

L'utilizzo di servizi come Amazon DynamoDB, Gateway Amazon API, bilanciatori del carico e AWS Lambda serverless avviene sfruttando il piano dati. Tuttavia, la creazione di nuove funzioni, bilanciatori del carico, gateway API o tabelle DynamoDB è un'azione del piano di controllo (control-

plane) e deve essere completata prima del deterioramento come preparazione a un evento e test delle azioni di failover. Per Amazon RDS, le azioni del piano dati consentono l'accesso ai dati.

Per ulteriori informazioni su piani dati, piani di controllo (control-plane) e su come AWS crea servizi per soddisfare gli obiettivi di alta disponibilità, consulta Stabilità statica con le zone di disponibilità.

Capire quali operazioni sono sul piano dati e quali sul piano di controllo (control-plane).

Passaggi dell'implementazione

Per ogni carico di lavoro che deve essere ripristinato dopo un evento di deterioramento, valuta il runbook di failover, il design ad alta disponibilità, il progetto di riparazione automatica o il piano di ripristino delle risorse HA. Identifica ogni azione che potrebbe essere considerata un'azione del piano di controllo (control-plane).

Prendi in considerazione la possibilità di modificare l'azione di controllo in un'azione del piano dati:

- Auto Scaling (piano di controllo) rispetto alle risorse predimensionate di Amazon EC2 (piano dati)
- Dimensionamento delle istanze Amazon EC2 (piano di controllo) sul dimensionamento AWS Lambda (piano dati)
- Valuta qualsiasi progetto utilizzando Kubernetes e considerando la natura delle azioni del piano di controllo (control-plane). L'aggiunta di pod è un'azione del piano dati in Kubernetes. Le azioni devono limitarsi all'aggiunta di pod e non all'aggiunta di nodi. L'utilizzo di nodi con provisioning eccessivo è il metodo preferibile per limitare le azioni del piano di controllo (control-plane).

Prendi in considerazione approcci alternativi che consentano alle azioni del piano dati di incidere sulla stessa correzione.

- Modifica dei record di Route 53 (piano di controllo (control-plane) o Amazon Application Recovery Controller (piano dati)
- Controlli dell'integrità di Route 53 per aggiornamenti più automatizzati

Se il servizio è mission critical, prendi in considerazione alcuni servizi in una regione secondaria per consentire più azioni del piano di controllo (control-plane) e del piano dati in una regione non interessata dal problema.

- Amazon EC2 Auto Scaling o Amazon EKS in una regione primaria rispetto ad Amazon EC2 Auto Scaling o Amazon EKS in una regione secondaria e instradamento del traffico verso una regione secondaria (azione del piano di controllo (control-plane))

- Crea una replica di lettura nella regione secondaria o tenta la stessa azione nella regione principale (azione del piano di controllo (control-plane))

Risorse

Best practice correlate:

- Definizione di disponibilità
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti

Documenti correlati:

- Partner APN: partner che possono essere d'aiuto con l'automazione della tua tolleranza ai guasti
- Marketplace AWS: prodotti utilizzabili per la tolleranza ai guasti
- Amazon Builders' Library: Avoiding overload in distributed systems by putting the smaller service in control
- API Amazon DynamoDB (piano di controllo (control-plane) e piano dati)
- AWS Lambda Executions (suddivise in piano di controllo (control-plane) e piano dati)
- AWS Elemental MediaStore Data Plane
- Building highly resilient applications using Amazon Application Recovery Controller, Part 1: Single-Region stack
- Building highly resilient applications using Amazon Application Recovery Controller, Part 2: Multi-Region stack
- Creating Disaster Recovery Mechanisms Using Amazon Route 53
- What is Amazon Application Recovery Controller
- Piano di controllo (control-plane) e piano dati di Kubernetes

Video correlati:

- Back to Basics - Using Static Stability
- Building resilient multi-site workloads using AWS global services

Esempi correlati:

- Introducing Amazon Application Recovery Controller

- Amazon Builders' Library: Avoiding overload in distributed systems by putting the smaller service in control
- Building highly resilient applications using Amazon Application Recovery Controller, Part 1: Single-Region stack
- Building highly resilient applications using Amazon Application Recovery Controller, Part 2: Multi-Region stack
- Stabilità statica con le zone di disponibilità

Strumenti correlati:

- Amazon CloudWatch
- AWS X-Ray

REL11-BP05 Utilizzo della stabilità statica per evitare un comportamento bimodale

I carichi di lavoro devono essere staticamente stabili e funzionare in una singola modalità normale. Il comportamento bimodale si verifica quando il carico di lavoro presenta un comportamento diverso in modalità normale e in modalità di guasto.

Ad esempio, ciò potrebbe accadere nel momento in cui si prova a ripristinare un guasto nella zona di disponibilità avviando nuove istanze in una zona di disponibilità diversa. Questo approccio può comportare una risposta bimodale durante una modalità di guasto. È invece necessario creare carichi di lavoro che siano staticamente stabili e operino in una sola modalità. In questo esempio, le nuove istanze avrebbero dovuto essere allocate nella seconda zona di disponibilità già prima del guasto. Questo design staticamente stabile verifica che il carico di lavoro funzioni in una sola modalità.

Risultato desiderato: i carichi di lavoro non presentano un comportamento bimodale in modalità normale e in modalità di guasto.

Anti-pattern comuni:

- Supporre che le risorse possano sempre essere allocate indipendentemente dall'ambito del guasto.
- Tentare di acquisire risorse in modo dinamico durante un guasto.
- Non rendere disponibili risorse adeguate tra zone o regioni diverse fino a quando non si verifica un guasto.

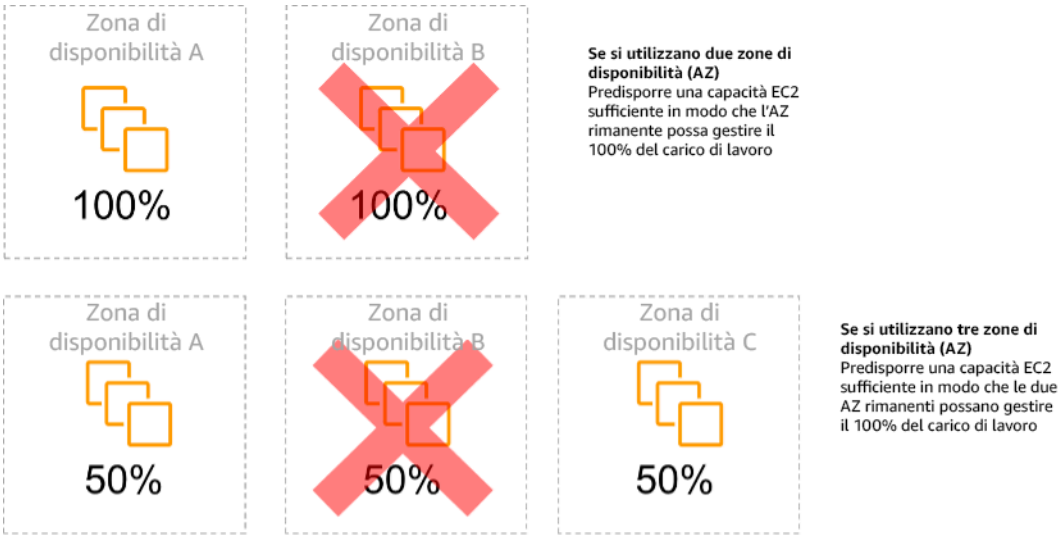
- Considerare i progetti staticamente stabili solo per risorse di calcolo.

Vantaggi dell'adozione di questa best practice: i carichi di lavoro eseguiti con progetti staticamente stabili sono in grado di avere risultati prevedibili durante eventi normali e di guasto.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Il comportamento bimodale ha luogo quando il carico di lavoro mostra un comportamento diverso in modalità normale e di guasto, ad esempio facendo affidamento sull'avvio di nuove istanze se una zona di disponibilità presenta un malfunzionamento. Un esempio di comportamento bimodale è quello che si verifica quando progetti Amazon EC2 stabili rendono disponibili un numero sufficiente di istanze in ciascuna zona di disponibilità per gestire il carico di lavoro in caso di rimozione di una di tali zone. Elastic Load Balancing o Amazon Route 53 controllano lo stato in modo da trasferire il carico dalle istanze danneggiate. Dopo il trasferimento del traffico, è possibile utilizzare AWS Auto Scaling per sostituire in modo asincrono le istanze della zona interessata dal guasto avviandole nelle zone integre. La stabilità statica per l'implementazione delle risorse di calcolo (ad esempio istanze EC2 o container) determinerà la massima affidabilità.



Stabilità statica delle istanze EC2 nelle diverse zone di disponibilità

Questo approccio deve essere valutato rispetto al costo associato al modello e al valore aziendale attribuito al mantenimento della disponibilità del carico di lavoro in tutti i casi di resilienza. Fornire una minore capacità di elaborazione e affidarsi all'avvio di nuove istanze in caso di guasto è meno costoso. Tuttavia, in caso di guasti su larga scala, come una zona di disponibilità o un problema

a livello regionale, tale approccio è meno efficace, perché si basa su un piano operativo e sulla disponibilità di risorse sufficienti nelle zone o nelle regioni non interessate dal problema.

La soluzione deve inoltre valutare l'affidabilità rispetto ai costi necessari per il carico di lavoro. Gli approcci che garantiscono la stabilità statica si applicano a una varietà di architetture, tra cui istanze di calcolo distribuite tra zone di disponibilità, progetti di repliche di lettura di database, progetti di cluster Kubernetes (Amazon EKS) e architetture di failover multiregione.

È anche possibile implementare un progetto staticamente più stabile utilizzando più risorse in ciascuna zona. Aggiungendo più zone, si riduce la quantità di elaborazione aggiuntiva necessaria per la stabilità statica.

Un altro esempio di comportamento bimodale potrebbe derivare da un timeout di rete in grado di causare un tentativo di aggiornamento dello stato di configurazione dell'intero sistema. Ciò potrebbe aggiungere un carico imprevisto su un altro componente che potrebbe quindi generare un errore, innescando ulteriori conseguenze impreviste. Questo loop di feedback negativo influisce sulla disponibilità del carico di lavoro. Al contrario, è possibile creare sistemi che siano staticamente stabili e funzionino in una sola modalità. Un progetto staticamente stabile potrebbe eseguire con continuità un'attività e aggiornare sempre, con cadenza regolare, lo stato della configurazione. Quando una chiamata non va a buon fine, il carico di lavoro può utilizzare il valore precedentemente memorizzato nella cache e segnalare un allarme.

Un altro esempio di comportamento bimodale è consentire ai client di bypassare la cache del carico di lavoro quando si verificano guasti. Potrebbe sembrare una soluzione che soddisfa le esigenze del client, ma non dovrebbe essere consentita perché modifica in modo significativo le richieste sul carico di lavoro e potrebbe causare dei guasti.

Valuta i carichi di lavoro critici per determinare quali carichi di lavoro richiedono questo tipo di progettazione di resilienza. Per quelli considerati critici, deve essere esaminato ogni componente dell'applicazione. Alcuni tipi di servizi che richiedono valutazioni di stabilità statica sono:

- Calcolo: Amazon EC2, EKS-EC2, ECS-EC2, EMR-EC2
- Database: Amazon Redshift, Amazon RDS, Amazon Aurora
- Archiviazione: Amazon S3 (zona singola), Amazon EFS (supporti), Amazon FSx (supporti)
- Bilanciatori del carico: in base a determinati progetti

Passaggi dell'implementazione

- Realizzare sistemi che siano staticamente stabili e operino in una sola modalità. In questo caso, effettuare il provisioning di un numero sufficiente di istanze in ogni zona o regione di disponibilità per gestire la capacità del carico di lavoro qualora venga rimossa una zona o regione di disponibilità. Per l'indirizzamento verso risorse integre è possibile utilizzare una varietà di servizi, come:
  - Instradamento DNS tra più regioni
  - Instradamento tra più regioni MRAP per Amazon S3
  - AWS Global Accelerator
  - Amazon Application Recovery Controller
- Configura repliche di lettura del database per tenere conto della perdita di una singola istanza primaria o di una replica di lettura. Se il traffico viene servito da repliche di lettura, la quantità in ogni zona di disponibilità e in ogni regione deve corrispondere al fabbisogno complessivo in caso di guasto della zona o della regione.
- Configurare i dati critici nel sistema di archiviazione Amazon S3 progettato per essere staticamente stabile rispetto ai dati archiviati in caso di guasto della zona di disponibilità. In caso di utilizzo della classe di archiviazione Amazon S3 One Zone-IA, questa non deve essere considerata staticamente stabile, poiché la perdita di tale zona riduce al minimo l'accesso ai dati archiviati.
- I bilanciatori del carico sono a volte configurati in modo errato o sono progettati per servire una zona di disponibilità specifica. In questo caso, il progetto staticamente stabile potrebbe consistere nel distribuire un carico di lavoro su più zone di disponibilità seguendo un design più complesso. Il progetto originale potrebbe essere utilizzato per ridurre il traffico tra zone per motivi di sicurezza, latenza o costi.

Risorse

Best practice Well-Architected correlate:

- Definizione di disponibilità
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP04 Fare affidamento al piano dati invece che al piano di controllo (control-plane) durante il ripristino

Documenti correlati:

- Minimizing Dependencies in a Disaster Recovery Plan
- The Amazon Builders' Library: stabilità statica con le zone di disponibilità
- Limiti di isolamento dei guasti
- Stabilità statica con le zone di disponibilità
- Multi-zona RDS
- Minimizing Dependencies in a Disaster Recovery Plan
- Instradamento DNS tra più regioni
- Instradamento tra più regioni MRAP per Amazon S3
- AWS Global Accelerator
- Amazon Application Recovery Controller
- Amazon S3 a zona singola
- Bilanciamento del carico su più zone

Video correlati:

- Static stability in AWS: AWS re:Invent 2019: Introducing The Amazon Builders' Library (DOP328)

REL11-BP06 Invio di notifiche quando gli eventi influiscono sulla disponibilità

Le notifiche vengono inviate al rilevamento del superamento delle soglie, anche se l'evento causato dal problema è stato risolto automaticamente.

Il ripristino automatizzato consente al carico di lavoro di risultare affidabile. Tuttavia, potrebbe anche nascondere problemi sottostanti che devono essere risolti. Implementa il monitoraggio e gli eventi appropriati in modo da poter rilevare i modelli di problemi, inclusi quelli risolti dalla diagnostica automatica e risolvere così i problemi della causa principale.

I sistemi resilienti sono progettati in modo che gli eventi di degrado vengano immediatamente comunicati ai team appropriati. Queste notifiche devono essere inviate tramite uno o più canali di comunicazione.

Risultato desiderato: gli avvisi vengono inviati immediatamente ai team operativi quando vengono superate soglie come i tassi di errore, la latenza o altri parametri critici degli indicatori chiave di prestazione (KPI), in modo che questi problemi vengano risolti il prima possibile e l'impatto sugli utenti sia evitato o ridotto al minimo.

Anti-pattern comuni:

- Invio di un numero eccessivo di allarmi.
- Invio di allarmi non utilizzabili.
- Impostazione di soglie di allarme troppo alte (troppo sensibili) o troppo basse (troppo poco sensibili).
- Mancato invio di allarmi per dipendenze esterne.
- Mancata presa in considerazione dei guasti nell'area grigia nella progettazione di sistemi di monitoraggio e allarmi.
- Eseguire l'automazione del risanamento, ma senza avvisare il team competente che era necessario un intervento di ripristino.

Vantaggi dell'adozione di questa best practice: le notifiche di ripristino rendono i team operativi e aziendali consapevoli dei peggioramenti del servizio in modo che possano reagire immediatamente per ridurre al minimo sia il tempo medio di rilevamento (MTTD) che il tempo medio di riparazione (MTTR). Le notifiche degli eventi di ripristino consentono anche di non ignorare i problemi che si verificano di rado.

Livello di rischio associato se questa best practice non fosse adottata: medio. La mancata implementazione di meccanismi di monitoraggio e notifica degli eventi appropriati può comportare l'impossibilità di rilevare i modelli di problemi, compresi quelli risolti mediante la correzione automatica. Un team verrà informato del degrado del sistema solo nel momento in cui gli utenti contattano il servizio clienti o per caso.

Guida all'implementazione

Quando si definisce una strategia di monitoraggio, un allarme attivato è un evento comune. Questo evento dovrebbe contenere un identificatore dell'allarme, lo stato dell'allarme (ad esempio `IN ALARM` o 0K) e i dettagli di ciò che lo ha attivato. In molti casi, è necessario rilevare un evento di allarme e inviare una notifica tramite e-mail. Questo è un esempio di operazione su un allarme. La notifica degli allarmi è fondamentale per l'osservabilità, in quanto informa le persone giuste della presenza di un problema. Tuttavia, quando le operazioni eseguite sulla base degli eventi raggiungono un certo grado di maturità nella soluzione di osservabilità, è possibile risolvere automaticamente il problema senza la necessità dell'intervento umano.

Una volta stabiliti gli allarmi di monitoraggio dei KPI, è necessario inviare avvisi ai team appropriati quando vengono superate le soglie. Tali avvisi possono essere utilizzati anche per attivare processi automatizzati che tenteranno di porre rimedio al danno o alla compromissione.

Per un monitoraggio delle soglie più complesso, è necessario prendere in considerazione gli allarmi compositi. Gli allarmi compositi utilizzano una serie di allarmi di monitoraggio dei KPI per creare un avviso basato sulla logica di business operativa. Gli allarmi CloudWatch possono essere configurati per l'invio di e-mail o per la registrazione di eventi imprevisti in sistemi di monitoraggio degli eventi imprevisti di terze parti tramite l'integrazione con Amazon SNS o Amazon EventBridge.

Passaggi dell'implementazione

Crea vari tipi di allarmi in base al modo in cui vengono monitorati i carichi di lavoro, ad esempio:

- Gli allarmi applicativi vengono utilizzati per rilevare quando una parte del carico di lavoro non funziona correttamente.
- Gli allarmi infrastrutturali indicano quando scalare le risorse. È possibile mostrare visivamente gli allarmi sui pannelli di controllo, inviarli tramite Amazon SNS o e-mail e utilizzarli con Auto Scaling per ridurre orizzontalmente o aumentare orizzontalmente i carichi di lavoro.
- È possibile creare semplici allarmi statistici per monitorare quando una metrica supera una soglia statica per un numero specificato di periodi di valutazione.
- Gli allarmi compositi possono tenere conto di allarmi complessi provenienti da più fonti.
- Una volta creato l'allarme è possibile generare eventi di notifica appropriati. Puoi richiamare direttamente un'API Amazon SNS per inviare notifiche e collegare qualsiasi automazione ai fini della riparazione o della comunicazione.
- Con AWS Health si ricevono le informazioni sul degrado delle prestazioni del servizio. Si creano notifiche di eventi AWS Health personalizzati per i canali e-mail e chat con Notifiche all'utente AWS e si usano gli strumenti di monitoraggio e avviso con Amazon EventBridge per l'integrazione a livello di codice.

Risorse

Best practice Well-Architected correlate:

- Definizione di disponibilità

Documenti correlati:

- Creating a CloudWatch Alarm Based on a Static Threshold
- What Is Amazon EventBridge?
- What is Amazon Simple Notification Service?
- Publishing Custom Metrics
- Using Amazon CloudWatch Alarms
- Configurazione degli allarmi compositi di CloudWatch
- What's new in AWS Observability at re:Invent 2022

Strumenti correlati:

- CloudWatch
- CloudWatch X-Ray

REL11-BP07 Progettazione del prodotto in modo da soddisfare gli obiettivi di disponibilità e gli accordi sul livello di servizio (SLA) per i tempi di attività

Progetta il tuo prodotto per soddisfare gli accordi sul livello di servizio (SLA) sul tempo di attività e sugli obiettivi di disponibilità. Se pubblichi o accetti privatamente obiettivi di disponibilità o contratti sul livello di servizio per i tempi di attività, verifica che l'architettura e i processi operativi siano progettati in modo da supportarli.

Risultato desiderato: ogni applicazione presenta un obiettivo definito in termini di disponibilità e uno SLA per le metriche delle prestazioni, monitorabili e gestibili per soddisfare i risultati aziendali.

Anti-pattern comuni:

- Progettazione e implementazione di carichi di lavoro senza predisporre alcun SLA.
- Impostazione di metriche elevate per lo SLA senza fondamento logico o requisiti aziendali.
- Impostazione di SLA senza tenere conto delle dipendenze e dei relativi SLA sottostanti.
- Progettazione delle applicazioni senza tenere conto del Modello di responsabilità condivisa per la resilienza.

Vantaggi dell'adozione di questa best practice: soddisfare gli obiettivi aziendali e le aspettative dei clienti grazie alla progettazione di applicazioni in base a obiettivi chiave in termini di resilienza.

Questi obiettivi orientano un processo di progettazione delle applicazioni in grado di valutare diverse tecnologie e tenere conto di vari compromessi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

La progettazione delle applicazioni deve tenere conto di una serie eterogenea di requisiti derivati da obiettivi aziendali, operativi e finanziari. Nell'ambito dei requisiti operativi, i carichi di lavoro devono avere obiettivi specifici in termini di metriche di resilienza, in modo da poter essere monitorati e supportati correttamente. Le metriche di resilienza non devono essere impostate o derivate dopo l'implementazione del carico di lavoro. Devono invece essere definite durante la fase di progettazione e contribuire a determinare i diversi compromessi e decisioni.

- Ogni carico di lavoro deve avere una serie di metriche di resilienza propria. Le metriche possono essere diverse da quelle di altre applicazioni aziendali.
- La riduzione delle dipendenze può avere un impatto positivo sulla disponibilità. Per ogni carico di lavoro è necessario considerare le dipendenze e i relativi SLA. In generale, seleziona dipendenze con obiettivi di disponibilità uguali o maggiori rispetto agli obiettivi del carico di lavoro.
- Prendi in considerazione progettazioni con accoppiamento debole in modo che il carico di lavoro possa funzionare correttamente anche in caso di dipendenze compromesse, se possibile.
- Riduci le dipendenze del piano di controllo (control-plane), in particolare durante un ripristino o un peggioramento delle prestazioni. Valuta le progettazioni staticamente stabili per carichi di lavoro mission critical. Usa il contenimento delle risorse per aumentare la disponibilità delle dipendenze in un carico di lavoro.
- L'osservabilità e la strumentazione sono essenziali per soddisfare i contratti sul livello di servizio attraverso la riduzione del tempo medio di rilevamento (MTTD) e del tempo medio di ripristino (MTTR).
- Errori meno frequenti (tempo medio tra guasti, o MTBF, più lungo), tempi di rilevamento degli errori più brevi (MTTD minore) e tempi di riparazione più brevi (MTTR minore) sono i tre fattori usati per migliorare la disponibilità in sistemi distribuiti.
- La definizione e l'applicazione di metriche di resilienza per un carico di lavoro sono essenziali per qualsiasi progettazione efficace. Queste progettazioni devono tenere conto dei compromessi introdotti dalla complessità di progettazione, delle dipendenze dei servizi, delle prestazioni, del dimensionamento e dei costi.

Passaggi dell'implementazione

- Esamina e documenta la progettazione del carico di lavoro cercando di rispondere alle domande seguenti:
  - Dove vengono usati i piani di controllo (control-plane) nel carico di lavoro?
  - Come viene implementata la tolleranza ai guasti nel carico di lavoro?
  - Quali sono i modelli di progettazione per dimensionamento, scalabilità automatica, ridondanza e componenti a disponibilità elevata?
  - Quali sono i requisiti per la disponibilità e la coerenza dei dati?
  - Vi sono aspetti da considerare in fatto di contenimento delle risorse o stabilità statica delle risorse?
  - Quali sono le dipendenze dei servizi?
- Definisci insieme alle parti interessate le metriche per lo SLA in base all'architettura del carico di lavoro. Tieni conto degli SLA di tutte le dipendenze usate dal carico di lavoro.
- Una volta definiti gli obiettivi dello SLA, ottimizza l'architettura in modo da soddisfarlo.
- Una volta impostata una progettazione che soddisfa lo SLA, implementa modifiche operative, automazione dei processi e runbook anch'essi incentrati sulla riduzione dell'MTTD e dell'MTTR.
- Dopo aver implementato lo SLA, devi monitorarlo e documentarlo.

Risorse

Best practice correlate:

- REL03-BP01 Scegli come segmentare il tuo carico di lavoro
- REL10-BP01 Implementazione del carico di lavoro in diversi luoghi
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL11-BP03 Automatizzazione della riparazione a tutti i livelli
- REL12-BP04 Test della resilienza tramite l’utilizzo dell’ingegneria del caos
- REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati
- Comprendere lo stato del carico di lavoro

Documenti correlati:

- Availability with redundancy
- Pilastro dell'affidabilità: disponibilità

- Measuring availability
- Limiti di isolamento dei guasti di AWS
- Modello di responsabilità condivisa per la resilienza
- Stabilità statica con le zone di disponibilità
- Contratti sul livello di servizio (SLA) AWS
- Guidance for Cell-based Architecture on AWS
- AWS infrastructure
- Whitepaper Advanced Multi-AZ Resiliance Patterns

Servizi correlati:

- Amazon CloudWatch
- AWS Config
- AWS Trusted Advisor

REL 12. Come si testa l'affidabilità?

Dopo aver progettato il carico di lavoro in modo da essere resiliente alle sollecitazioni della produzione, i test sono l'unico modo per verificare il funzionamento corretto e offrire la resilienza prevista.

Best practice

- REL12-BP01 Utilizzo dei playbook per analizzare gli errori
- REL12-BP02 Esecuzione di analisi post-incidente
- REL12-BP03 Test dei requisiti di scalabilità e prestazioni
- REL12-BP04 Test della resilienza tramite l'utilizzo dell'ingegneria del caos
- REL12-BP05 Esecuzione regolare di GameDay

REL12-BP01 Utilizzo dei playbook per analizzare gli errori

Consenti risposte coerenti e tempestive a scenari di guasto che non sono ben compresi, documentando il processo di analisi nei playbook. I playbook sono le fasi predefinite eseguite per identificare i fattori che contribuiscono a uno scenario di guasto. I risultati provenienti da un

passaggio del processo vengono utilizzati per stabilire i passaggi successivi da intraprendere fino all'identificazione o alla risoluzione del problema.

Il playbook è una pianificazione proattiva che è necessario eseguire, in modo da potere intraprendere azioni reattive in modo efficace. Se durante la produzione si verificano scenari di guasto non coperti dal playbook, risolvi innanzitutto il problema (spegni l'incendio). Quindi torna indietro e osserva le fasi intraprese per risolvere il problema e utilizzale per aggiungere una nuova voce al playbook.

Tieni presente che i playbook vengono utilizzati in risposta a specifici incidenti, mentre i runbook vengono utilizzati per ottenere esiti specifici. Spesso, i runbook vengono utilizzati per le attività di routine e i playbook vengono utilizzati per rispondere a eventi non di routine.

Anti-pattern comuni:

- Pianificare la distribuzione di un carico di lavoro senza conoscere i processi per diagnosticare i problemi o rispondere agli incidenti.
- Decisioni non pianificate sui sistemi da cui raccogliere log e parametri durante l'analisi di un evento.
- Non conservare parametri e eventi abbastanza a lungo da poter recuperare i dati.

Vantaggi dell'adozione di questa best practice: l'acquisizione dei playbook garantisce l'esecuzione coerente dei processi. La codifica dei playbook limita l'introduzione di errori derivanti dall'attività manuale. L'automazione dei playbook riduce il tempo necessario per rispondere a un evento eliminando il requisito per l'intervento dei membri del team o fornendo loro informazioni aggiuntive quando inizia l'intervento.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

- Utilizza playbook per identificare i problemi. I playbook sono processi documentati per eseguire indagini sui problemi. Promuovi risposte coerenti e tempestive agli scenari di errore documentando i processi nei playbook. I playbook devono contenere le informazioni e le istruzioni necessarie affinché una persona adeguatamente qualificata possa raccogliere le informazioni applicabili, identificare potenziali fonti di errore, isolare i guasti e stabilire i fattori che contribuiscono all'origine di un problema (eseguire l'analisi post-incidente).
  - Implementazione dei playbook come codice. Esegui le operazioni come codice mediante lo scripting dei playbook per assicurare coerenza e ridurre gli errori causati dai processi manuali. I playbook possono essere composti da più script che rappresentano le diverse fasi

che potrebbero essere necessarie per identificare i fattori che contribuiscono all'origine di un problema. Le attività dei runbook possono essere richiamate o eseguite nell'ambito delle attività dei playbook oppure possono richiedere l'esecuzione di un playbook in risposta agli eventi identificati.

- Automatizza i playbook operativi con AWS Systems Manager
- AWS Systems Manager Run Command
- AWS Systems Manager Automation
- Che cos'è AWS Lambda?
- What Is Amazon EventBridge?
- Using Amazon CloudWatch Alarms

Risorse

Documenti correlati:

- AWS Systems Manager Automation
- AWS Systems Manager Run Command
- Automatizza i playbook operativi con AWS Systems Manager
- Using Amazon CloudWatch Alarms
- Utilizzo di Canary (Amazon CloudWatch Synthetics)
- What Is Amazon EventBridge?
- Che cos'è AWS Lambda?

Esempi correlati:

- Automazione delle operazioni con playbook e runbook

REL12-BP02 Esecuzione di analisi post-incidente

Esamina gli eventi che influiscono sui clienti e identifica i fattori che vi hanno contribuito e gli elementi di azione preventivi. Utilizza queste informazioni per sviluppare modi per limitare o prevenire il ripetersi degli incidenti. Sviluppa procedure per attivare risposte rapide ed efficaci. Comunica i fattori che hanno contribuito al presentarsi dell'imprevisto e le azioni correttive secondo necessità, specificamente mirate per il pubblico di destinazione. All'occorrenza, adotta un metodo per comunicare queste cause ad altri.

Valuta perché i test esistenti non hanno individuato il problema. Aggiungi i test per questo caso se i test non esistono già.

Risultato desiderato: i tuoi team dispongono di un approccio coerente e concordato per la gestione dell'analisi post-incidente. Un meccanismo è il processo di correzione dell'errore (COE). Il processo COE aiuta i team a individuare, comprendere e gestire le cause principali degli incidenti, creando al contempo meccanismi e guardrail per limitare la probabilità che lo stesso incidente si ripeta.

Anti-pattern comuni:

- Individuare i fattori che hanno contribuito al verificarsi dell'incidente, ma non continuare a cercare in maniera più approfondita altri potenziali problemi e approcci da mitigare.
- Identificare le cause degli errori umani senza fornire alcuna formazione o automazione che potrebbe prevenirli.
- Concentrarsi sull'attribuzione delle colpe piuttosto che sulla comprensione della causa principale, creando così una cultura della paura e ostacolando la comunicazione costruttiva
- Mancata condivisione delle informazioni, che mantiene gli esiti dell'analisi degli incidenti all'interno di un gruppo ristretto e impedisce ad altri di beneficiare delle lezioni apprese
- Nessun meccanismo che consenta di acquisire le conoscenze formali; in questo modo si perdono informazioni preziose in quanto non vengono preservate le lezioni apprese sotto forma di best practice aggiornate, con il conseguente rischio che gli incidenti si ripetano con la stessa causa principale o causa simile

Vantaggi dell'adozione di questa best practice: l'esecuzione di analisi post-incidente e la condivisione dei risultati consente ad altri carichi di lavoro di mitigare il rischio se hanno implementato gli stessi fattori che hanno contribuito al verificarsi dell'incidente e permette loro di implementare la mitigazione o il ripristino automatico prima che si verifichi un incidente.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Una buona analisi post-incidente fornisce opportunità per proporre soluzioni comuni a problemi con modelli di architettura utilizzati in altri punti nei tuoi sistemi.

Un elemento fondamentale del processo COE è la documentazione e la risoluzione dei problemi. È consigliabile definire un modo standard per documentare le cause principali critiche e assicurarsi che queste vengano esaminate e risolte. Assegna in modo chiaro il responsabile del processo di analisi

post-incidente. Nomina un team o una persona responsabile della supervisione delle indagini e dei follow-up degli incidenti.

Promuovi una cultura basata sull'apprendimento e sul miglioramento piuttosto che sull'attribuzione di colpe. Insisti sul fatto che l'obiettivo è prevenire incidenti futuri e non penalizzare le persone.

Sviluppa procedure ben definite per l'esecuzione delle analisi post-incidente. Queste procedure dovrebbero stabilire le misure da adottare, le informazioni da raccogliere e le questioni chiave da risolvere durante l'analisi. Svolgi indagini approfondite sugli incidenti, andando oltre le cause immediate per identificare le cause principali e i fattori determinanti. Utilizza tecniche come i Cinque Perché per analizzare in modo approfondito i problemi sottostanti.

Mantieni un archivio delle conclusioni derivanti dalle analisi degli incidenti. Queste conoscenze formali possono fungere da riferimento per futuri incidenti e attività di prevenzione. Condividi gli esiti e gli approfondimenti delle analisi post-incidente e valuta la possibilità di organizzare riunioni di revisione post-incidente con invito aperto per discutere i risultati e le conclusioni.

Passaggi dell'implementazione

- Durante l'analisi post-incidente, assicurati che il processo non comporti la colpevolizzazione delle parti coinvolte. Ciò consente alle parti interessate di essere imparziali rispetto delle azioni correttive proposte, nonché di promuovere l'autovalutazione e la collaborazione a livello di team.
- Definisci una procedura standardizzata per documentare i problemi critici. Una struttura di esempio per tale documento è la seguente:
  - Che cos'è successo?
  - Quale impatto ha avuto su clienti e attività?
  - Qual è stata la causa principale?
  - Di quali dati disponi a supporto di ciò?
    - Ad esempio, metriche e grafici
  - Quali sono state le principali implicazioni sui pilastri critici, specialmente per quanto riguarda la sicurezza?
    - Quando progetti l'architettura dei carichi di lavoro, devi trovare dei compromessi tra i pilastri su cui si regge il contesto aziendale. Le decisioni aziendali possono stabilire le priorità di progettazione. Potresti ridurre i costi a spese dell'affidabilità in ambienti di sviluppo oppure, per quanto riguarda le soluzioni mission-critical, potresti ottimizzare l'affidabilità con costi maggiori. La sicurezza ha la massima priorità quando si tratta di proteggere i tuoi clienti.
  - Quali lezioni hai imparato?

    - Quali azioni correttive stai adottando?
        - Elementi d'azione
        - Voci correlate
- Crea precise procedure operative standard per lo svolgimento delle analisi post-incidente.
- Configura un processo standardizzato di segnalazione degli incidenti. Documenta in modo esaustivo tutti gli incidenti, includendo il rapporto iniziale sull'incidente, i log, le comunicazioni e le azioni intraprese durante l'incidente.
- Ricorda che un incidente non necessariamente comporta un'interruzione del servizio. Potrebbe trattarsi di un near miss o di un sistema che funziona in modo imprevisto pur continuando a svolgere la sua funzione aziendale.
- Migliora continuamente il processo di analisi post-incidente sulla base dei feedback e delle lezioni apprese.
- Acquisisci gli esiti chiave in un sistema di gestione delle conoscenze e valuta eventuali modelli da aggiungere alle linee guide per gli sviluppatori o alle liste di controllo usate nella fase di pre-implementazione.

Risorse

Documenti correlati:

- Why you should develop a correction of error (COE)

Video correlati:

- Amazon's approach to failing successfully
- AWS re:Invent 2021 - Amazon Builders' Library: Operational Excellence at Amazon

REL12-BP03 Test dei requisiti di scalabilità e prestazioni

Utilizza tecniche come i test di carico per convalidare che il carico di lavoro soddisfi i requisiti di dimensionamento e prestazioni.

Nel cloud puoi creare un ambiente di test su scala di produzione per il carico di lavoro su richiesta. Invece di affidarti a un ambiente di test con risorse ridotte verticalmente, che potrebbe portare a previsioni imprecise dei comportamenti in produzione, puoi utilizzare il cloud per fornire un ambiente

di test che rispecchi fedelmente l'ambiente di produzione previsto. Questo ambiente consente di eseguire i test in una simulazione più precisa delle condizioni reali in cui si trova l'applicazione.

Oltre ai test sulle prestazioni, è essenziale verificare che le risorse di base, le impostazioni di dimensionamento, Service Quotas e la progettazione di resilienza funzionino come previsto sotto carico. Questo approccio olistico verifica che l'applicazione sia in grado di scalare in modo affidabile e funzionare come richiesto, anche nelle condizioni più difficili.

Risultato desiderato: il carico di lavoro mantiene il comportamento previsto anche quando è soggetto a picchi di carico. Affronti in modo proattivo tutti i problemi di prestazioni che possono verificarsi con la crescita e l'evoluzione dell'applicazione.

Anti-pattern comuni:

- Utilizzi ambienti di test che non corrispondono strettamente all'ambiente di produzione.
- Tratti il test di carico come un'attività separata e una tantum, anziché come una parte integrata della pipeline di integrazione continua (CI) dell'implementazione.
- Non definisci requisiti di prestazione chiari e misurabili, come tempi di risposta, throughput e obiettivi di scalabilità.
- Esegui test con scenari di carico non realistici o insufficienti e non esegui test per picchi di carico, picchi improvvisi e carico elevato sostenuto.
- Non solleciti il carico di lavoro superando i limiti di carico previsti.
- Utilizzi strumenti di test di carico e di profiling delle prestazioni inadeguati o inappropriati.
- Non disponi di sistemi di monitoraggio e di avviso completi per monitorare le metriche delle prestazioni e rilevare anomalie.

Vantaggi dell'adozione di questa best practice:

- I test di carico aiutano a identificare i potenziali colli di bottiglia delle prestazioni del sistema prima del passaggio in produzione. Quando simuli il traffico e i carichi di lavoro a livello di produzione, puoi identificare le aree in cui il sistema può avere difficoltà a gestire il carico, come tempi di risposta lenti, vincoli delle risorse o errori di sistema.
- Testando il sistema in varie condizioni di carico, puoi comprendere meglio i requisiti delle risorse necessari per supportare il carico di lavoro. Queste informazioni possono aiutarti a prendere decisioni informate sull'allocazione delle risorse e a prevenire l'eccesso o il difetto di provisioning di risorse.

- Per identificare i potenziali punti di errore, puoi osservare come si comporta il carico di lavoro in condizioni di carico elevato. Queste informazioni aiutano a migliorare l'affidabilità e la resilienza del carico di lavoro implementando meccanismi di tolleranza agli errori, strategie di failover e misure di ridondanza, a seconda dei casi.
- Identifichi e risolvi tempestivamente i problemi di prestazioni, evitando così le costose conseguenze di interruzioni del sistema, tempi di risposta lenti e utenti insoddisfatti.
- I dati dettagliati sulle prestazioni e le informazioni di profiling raccolte durante i test possono aiutarti a risolvere i problemi correlati alle prestazioni che potrebbero sorgere in produzione. Questo può portare a una risposta e a una risoluzione più rapida degli incidenti, riducendo l'impatto sugli utenti e sulle operazioni dell'organizzazione.
- In alcuni settori, il test proattivo delle prestazioni può aiutare il carico di lavoro a soddisfare gli standard di conformità, riducendo il rischio di sanzioni o problemi legali.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Il primo passo consiste nel definire una strategia di test completa che copra tutti gli aspetti dei requisiti di dimensionamento e prestazioni. Per iniziare, definisci chiaramente gli obiettivi di livello di servizio (SLO) del carico di lavoro in base alle esigenze aziendali, come il throughput, l'istogramma della latenza e il tasso di errore. Quindi, progetta una suite di test in grado di simulare vari scenari di carico che vanno dall'utilizzo medio a picchi improvvisi e a carichi di picco sostenuti, e verifica che il comportamento del carico di lavoro soddisfi gli SLO. Questi test devono essere automatizzati e integrati nella pipeline di integrazione e implementazione continua per individuare le regressioni delle prestazioni nelle prime fasi del processo di sviluppo.

Per testare efficacemente il dimensionamento e le prestazioni, investi negli strumenti e nell'infrastruttura corretti. Ciò include strumenti di test di carico in grado di generare un traffico utente realistico, strumenti di profiling delle prestazioni per identificare i colli di bottiglia e soluzioni di monitoraggio per tracciare le metriche chiave. È importante verificare che gli ambienti di test corrispondano fedelmente all'ambiente di produzione in termini di infrastrutture e condizioni ambientali, in modo da ottenere risultati il più possibile accurati. Per rendere più facile replicare e scalare in modo affidabile le configurazioni simili a quelle di produzione, utilizza infrastructure as code e le applicazioni basate su container.

I test di dimensionamento e sulle prestazioni sono un processo continuo, non un'attività una tantum. Implementa il monitoraggio completo e gli avvisi per monitorare le prestazioni dell'applicazione in

produzione e utilizza questi dati per perfezionare continuamente le strategie di test e gli sforzi di ottimizzazione. Analizza regolarmente i dati sulle prestazioni per identificare i problemi emergenti, testare nuove strategie di dimensionamento e implementare ottimizzazioni per migliorare l'efficienza e l'affidabilità dell'applicazione. Quando adotti un approccio iterativo e impari costantemente dai dati di produzione, puoi verificare che l'applicazione è in grado di adattarsi alle richieste variabili degli utenti e mantenere la resilienza e le prestazioni ottimali nel tempo.

Passaggi dell'implementazione

1. Stabilisci requisiti di prestazioni chiari e misurabili, come tempi di risposta, throughput e obiettivi di scalabilità. Questi requisiti devono essere basati sui modelli di utilizzo del carico di lavoro, sulle aspettative degli utenti e sulle esigenze aziendali.
2. Seleziona e configura uno strumento di test del carico in grado di simulare accuratamente i modelli di carico e il comportamento degli utenti nell'ambiente di produzione.
3. Per migliorare la precisione dei risultati dei test, configura un ambiente di test che corrisponde strettamente all'ambiente di produzione, comprese le condizioni dell'infrastruttura e dell'ambiente.
4. Crea una suite di test che copre un'ampia gamma di scenari, dai modelli di utilizzo medio ai picchi di carico, ai picchi rapidi e ai carichi elevati sostenuti. Integra i test nelle pipeline di implementazione e distribuzione continua per individuare regressioni delle prestazioni fin dalle prime fasi del processo di sviluppo.
5. Esegui test di carico per simulare il traffico reale degli utenti e capire come si comporta l'applicazione in diverse condizioni di carico. Per sollecitare l'applicazione, supera il carico previsto e osserva il suo comportamento, come il degrado dei tempi di risposta, l'esaurimento delle risorse o gli errori di sistema. Ciò aiuta a identificare il punto di rottura dell'applicazione e a informare le strategie di dimensionamento. Valuta la scalabilità del carico di lavoro aumentandolo progressivamente e misura l'impatto sulle prestazioni per identificare i limiti di dimensionamento e pianificare le esigenze di capacità future.
6. Implementa monitoraggio completo e avvisi per tracciare le metriche delle prestazioni, rilevare le anomalie e avviare azioni di dimensionamento o notifiche quando vengono superate le soglie.
7. Monitora e analizza costantemente i dati sulle prestazioni per identificare le aree di miglioramento. Itera le strategie di test e gli sforzi di ottimizzazione.

Risorse

Best practice correlate:

- REL01-BP04 Monitoraggio e gestione delle quote

- REL06-BP01 Monitoraggio di tutti i componenti per il carico di lavoro (generazione)
- REL06-BP03 Invio di notifiche (elaborazione e avvisi in tempo reale)

Documenti correlati:

- Load testing applications
- Test del carico distribuito su AWS
- Monitoraggio delle prestazioni delle applicazioni
- Amazon EC2 Testing Policy

Esempi correlati:

- Distributed Load Testing on AWS (GitHub)

Strumenti correlati:

- Profilatore Amazon CodeGuru
- Amazon CloudWatch RUM
- Apache JMeter
- K6
- Vegeta
- Hey
- ab
- wrk
- Test del carico distribuito su AWS

REL12-BP04 Test della resilienza tramite l'utilizzo dell'ingegneria del caos

Esegui regolarmente esperimenti di ingegneria del caos in ambienti di produzione o per quanto possibile ambienti analoghi per capire in che modo il sistema risponde a condizioni avverse.

Risultato desiderato:

La resilienza del carico di lavoro viene regolarmente verificata mediante l'applicazione dell'ingegneria del caos sotto forma di esperimenti di iniezione di guasti o di inserimento di carichi imprevisti, nonché

mediante il test della resilienza che convalida i comportamenti previsti noti del carico di lavoro durante un evento. Combina l'ingegneria del caos e i test della resilienza per verificare se il carico di lavoro è in grado di superare i guasti dei componenti ed eseguire il ripristino da interruzioni del servizio impreviste con un impatto minimo o nullo.

Anti-pattern comuni:

- Progettazione della resilienza, ma mancata verifica del funzionamento del carico di lavoro nel suo complesso in caso di errori.
- Mancata sperimentazione in scenari reali e con carichi previsti.
- Mancato trattamento degli esperimenti come codice o loro conservazione durante il ciclo di sviluppo.
- Mancata esecuzione degli esperimenti di ingegneria del caos sia nella pipeline CI/CD che esternamente alle implementazioni.
- Mancato utilizzo delle precedenti analisi post-incidente durante la determinazione degli errori su cui eseguire i test.

Vantaggi dell'adozione di questa best practice: l'introduzione di errori per verificare la resilienza del carico di lavoro consente di verificare che le procedure di ripristino della progettazione resiliente funzionerà se viene generato un vero e proprio errore.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

L'ingegneria del caos offre ai team la possibilità di continuare a inserire scenari di errore reali (simulazioni) in modo controllato a livello di fornitore di servizi, infrastruttura, carico di lavoro e componente con un impatto minimo o nullo per i clienti. Consente inoltre ai team di imparare dagli errori e osservare, misurare e migliorare la resilienza dei carichi di lavoro, nonché verificare l'attivazione degli avvisi e se tali avvisi vengono recapitati ai team se si verifica un evento definito.

Se applicata in modo continuativo, l'ingegneria del caos può mettere in evidenza i difetti del carico di lavoro che, se non risolti, possono avere ripercussioni negative sulla disponibilità e sulle operazioni.

> **Note**
>
> L'ingegneria del caos è la disciplina che sperimenta un sistema per creare fiducia nella capacità del sistema di affrontare condizioni turbolenti nella produzione. – Principles of Chaos Engineering

Se un sistema è in grado di sopportare queste interruzioni, l'esperimento di ingegneria del caos deve essere convertito in test automatico di regressione. In questo modo, gli esperimenti di ingegneria del caos devono essere eseguiti nell'ambito del ciclo di vita dello sviluppo dei sistemi (SDLC) e della pipeline CI/CD.

Per garantire che il carico di lavoro sia in grado di gestire un guasto del componente, esegui l'iniezione di eventi di errore reali durante l'esecuzione degli esperimenti. Ad esempio, esegui esperimenti relativi alla perdita di istanze Amazon EC2 o a eventi di failover delle istanze database Amazon RDS primario e quindi verifica che il carico di lavoro non sia stato compromesso oppure o che si stato interessato solo in minima parte. Utilizza una combinazione di errori dei componenti per simulare gli eventi che possono essere causati da un'interruzione del servizio in una zona di disponibilità.

Per gli errori a livello di applicazione, ad esempio gli arresti anomali, puoi iniziare utilizzando fattori di stress, ad esempio l'esaurimento della memoria o della CPU.

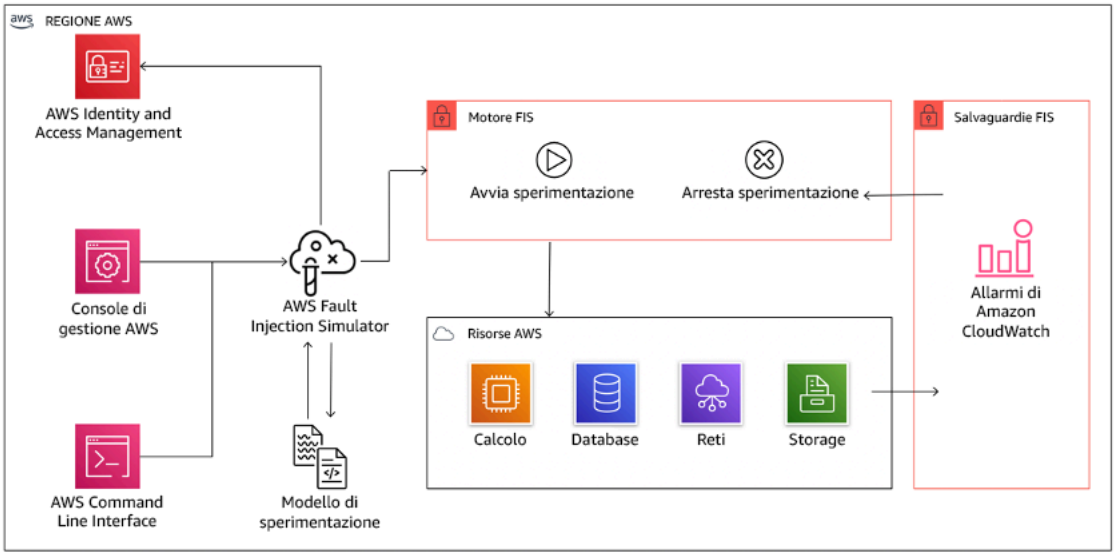
Per convalidare i meccanismi di fallback o failover per le dipendenze esterne causate da interruzioni intermittenti dei servizi di rete, i componenti devono simulare tale evento bloccando l'accesso ai fornitori di terze parti per una durata specificata, che può durare da pochi secondi ad alcune ore.

Altre modalità di degrado possono causare funzionalità ridotte e risposte lente, spesso con conseguente interruzione dei servizi. Le fonti comuni di questo degrado sono una maggiore latenza nei servizi critici e una comunicazione di rete inaffidabile (pacchetti persi). Gli esperimenti basati su questi errori, inclusi gli effetti a livello di rete come latenza, messaggi eliminati ed errori DNS, possono prevedere l'incapacità di risolvere un nome, raggiungere il servizio DNS o stabilire connessioni a servizi dipendenti.

Strumenti dell'ingegneria del caos:

AWS Fault Injection Service (AWS FIS) è un servizio completamente gestito per l'esecuzione di esperimenti di iniezione di guasti che possono essere utilizzati come parte della pipeline di CD o al suo esterno. AWS FIS è una soluzione estremamente valida da utilizzare durante i giorni di gioco

dell'ingegneria del caos. Supporta l'introduzione simultanea di errori su diversi tipi di risorse, tra cui Amazon EC2, Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS) e Amazon RDS. Questi errori includono la terminazione delle risorse, la forzatura dei failover, l'applicazione di fattori di stress a CPU o memoria, la limitazione (della larghezza di banda della rete), la latenza e la perdita di pacchetti. Poiché è integrato con gli allarmi Amazon CloudWatch, è possibile impostare condizioni di arresto come guardrail per eseguire il rollback di un esperimento se causa un impatto inatteso.



AWS Fault Injection Service è integrato con le risorse AWS per consentire l'esecuzione di esperimenti di iniezione di guasti per i carichi di lavoro.

Esistono anche diverse opzioni di terze parti per gli esperimenti di iniezione di guasti. Queste opzioni comprendono strumenti open source come Chaos Toolkit, Chaos Mesh e Litmus Chaos, nonché altre opzioni commerciali come Gremlin. Per ampliare l'ambito dei guasti che è possibile iniettare in AWS, AWS FIS si integra con Chaos Mesh e Litmus Chaos, così da coordinare i flussi di lavoro di iniezione di guasti tra più strumenti. Ad esempio, puoi eseguire un test di stress sulla CPU di un pod utilizzando gli errori di Chaos Mesh o Litmus Chaos durante la cessazione di una percentuale casualmente selezionata di nodi di cluster mediante le operazioni di errore di AWS FIS.

Passaggi dell'implementazione

1. Determinazione dei guasti da utilizzare per gli esperimenti.

   Valutazione della progettazione del carico di lavoro a livello di resilienza. Tali progettazioni (create seguendo le best practice del Framework Well-Architected) tengono conto dei rischi basati su dipendenze critiche, eventi passati, problemi noti e requisiti di conformità. Elenca i singoli

elementi della progettazione che devono conservare la resilienza e gli errori per mitigare i quali è stata sviluppata. Per ulteriori informazioni sulla creazione di tali elenchi, consulta il whitepaper sulla prontezza operativa, che illustra come creare un processo finalizzato alla prevenzione del ripetersi di incidenti precedenti. Il processo FMEA (Failure Modes and Effects Analysis) fornisce un framework per l'esecuzione di un'analisi degli errori a livello di componente e del relativo impatto sul carico di lavoro. Il processo FMEA è illustrato in maggiore dettaglio da Adrian Cockcroft in Failure Modes and Continuous Resilience.

2. Assegna una priorità a ogni errore.

   Comincia con una categorizzazione approssimativa, ad esempio alta, media o bassa. Per valutare la priorità, considera la frequenza dell'errore e l'impatto dell'errore sul carico di lavoro nel suo complesso.

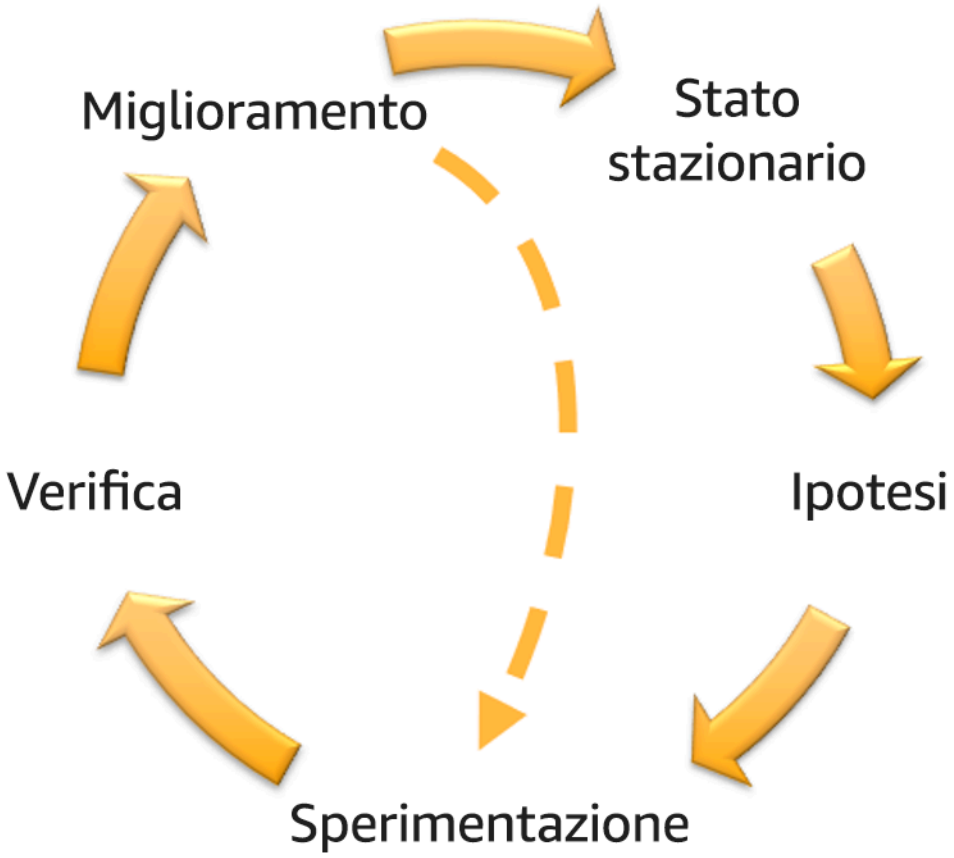
   Durante la valutazione della frequenza di un errore specifico, analizza i precedenti dati per lo stesso carico di lavoro, se disponibili. Se non sono disponibili, utilizza i dati di altri carichi di lavoro eseguiti in un ambiente simile.

   Durante la valutazione dell'impatto di un errore specifico, in genere maggiore è l'ambito dell'errore, maggiore sarà l'impatto. Considera la progettazione e lo scopo del carico di lavoro. Ad esempio, la capacità di accedere ai datastore di origine è di cruciale importanza per un carico di lavoro responsabile della trasformazione e dell'analisi dei dati. In questo caso, darai la precedenza agli esperimenti relativi agli errori di accesso, nonché a quelli con limitazione (della larghezza di banda della rete) e inserimento di latenza.

   Le analisi post-incidente rappresentano un'ottima fonte di dati per la comprensione della frequenza e dell'impatto delle modalità di errore.

   Utilizza la priorità assegnata per determinare il primo errore su cui eseguire l'esperimento e l'ordine in cui sviluppare i nuovi esperimenti di iniezione di guasti.

3. Per ogni esperimento eseguito, attieniti ai principi del volano dell'ingegneria del caos e della resilienza continua nella figura seguente.



Volano dell'ingegneria del caos e della resilienza continua, che utilizza il metodo scientifico di Adrian Hornsby.

a. Definisci lo stato stazionario come output misurabile di un carico di lavoro che indica un comportamento normale.

   Il carico di lavoro è associato allo stato stazionario se il suo funzionamento è affidabile e conforme a quanto previsto. Verifica pertanto che il carico di lavoro sia integro prima di definire lo stato stazionario. Lo stato stazionario non necessariamente indica l'assenza di impatto sul carico di lavoro se si verifica un errore in quanto una data percentuale di errori può rientrare nei limiti di valori accettabili. Lo stato stazionario rappresenta il punto di riferimento che verrà osservato durante l'esperimento e che metterà in evidenza le anomalie se le ipotesi definite nel passaggio successivo non sono conformi alle previsioni.

Ad esempio, lo stato stazionario di un sistema di pagamento può essere definito come elaborazione di 300 TPS con una percentuale di successo pari al 99% e un tempo di round trip pari a 500 ms.

b. Definisci un'ipotesi in merito alle reazioni del carico di lavoro all'errore.

Un'ipotesi ottimale fa riferimento al modo in cui il carico di lavoro presumibilmente è in grado di ridurre l'impatto dell'errore e salvaguardare lo stato stazionario. Nell'ipotesi è definito che, dato un errore di un tipo specifico, il sistema o il carico di lavoro rimarrà nello stato stazionario poiché la progettazione del carico di lavoro ha previsto sistemi specifici di attenuazione degli errori. Il tipo di errore specifico e i sistemi di attenuazione devono essere specificati nell'ipotesi.

Per l'ipotesi è possibile utilizzare il seguente modello, anche se è accettabile una formulazione diversa:

> **Note**
>
> Se si verifica un *`guasto specifico`*, il carico di lavoro *`nome del carico di lavoro illustrerà la mitigazione dei controlli`* per controbilanciare l'*`impatto sulle metriche aziendali o tecniche`*.

Esempio:

- In caso di arresto del 20% dei nodi nel gruppo di nodi Amazon EKS, l'API di creazione delle transazioni continua a servire il 99° percentile delle richieste in meno di 100 ms (stato stazionario). Verrà eseguito il ripristino dei nodi Amazon EKS entro cinque minuti; i pod verranno riprogrammati ed elaboreranno il traffico entro otto minuti dall'inizio dell'esperimento. Gli avvisi verranno attivati entro tre minuti.
- Se si verifica un errore in un'istanza Amazon EC2, il controllo dell'integrità di Elastic Load Balancing del sistema degli ordini farà sì che Elastic Load Balancing si limiti a inviare richieste alle rimanenti istanze integre, mentre Amazon EC2 Auto Scaling sostituirà l'istanza in errore, garantendo un incremento inferiore allo 0,01% degli errori (5xx) lato server (stato stazionario).
- Se l'istanza database primario Amazon RDS restituisce un errore, il carico di lavoro della raccolta di dati della catena di approvvigionamento eseguirà il failover e si connetterà all'istanza database in standby Amazon RDS per mantenere meno di un minuto di errori di lettura o scrittura del database (stato stazionario).

c. Esegui l'esperimento inserendo l'errore.

Per impostazione predefinita, un esperimento deve essere a prova di errore e tollerato dal carico di lavoro. Se sei consapevole del fatto che il carico di lavoro avrà esito negativo, non eseguire l'esperimento. L'ingegneria del caos deve essere utilizzata per individuare scenari noti sconosciuti o scenari completamente sconosciuti. Per scenari noti sconosciuti si intendono gli scenari di cui sei consapevole ma che non comprendi appieno, mentre scenari completamente sconosciuti si riferiscono a quegli scenari a te non noti e che non comprendi appieno. L'esecuzione di esperimenti su un carico di lavoro non funzionante non può fornire nuovi approfondimenti chiarificatori. L'esperimento deve infatti essere pianificato con attenzione, essere caratterizzato da un ambito ben definito relativamente al suo impatto, nonché fornire un meccanismo di rollback applicabile in caso di esiti negativi imprevisti. Se il criterio di due diligence indica che il carico di lavoro è in grado di sostenere l'esperimento, procedi ed esegui l'esperimento. Sono disponibili varie opzioni per l'inserimento degli errori. Per i carichi di lavoro su AWS, AWS FIS offre diverse simulazioni di guasto predefinite denominate operazioni. Puoi anche definire operazioni personalizzate eseguibili in AWS FIS utilizzando i documenti di AWS Systems Manager.

È sconsigliato l'uso di script personalizzati per gli esperimenti di ingegneria del caos, a meno che gli script non siano in grado di rilevare lo stato corrente del carico di lavoro, generare log e fornire meccanismi di rollback e condizioni di arresto, laddove possibile.

Un framework o set di strumenti efficace che supporta l'ingegneria del caos deve tenere traccia dello stato corrente di un esperimento, generare log e fornire meccanismi di rollback a supporto dell'esecuzione controllata di un esperimento. Inizia utilizzando un servizio noto, ad esempio AWS FIS, che consente di eseguire esperimenti con ambiti e meccanismi di sicurezza ben definiti in grado di eseguire il rollback dell'esperimento in caso di esiti negativi imprevisti. Per ulteriori informazioni su una varietà più ampia di esperimenti mediante AWS FIS, consulta anche il lab Resilient and Well-Architected Apps with Chaos Engineering. Inoltre, AWS Resilience Hub analizzerà il carico di lavoro e creerà gli esperimenti che potrai scegliere di implementare ed eseguire in AWS FIS.

> **Note**
>
> Per ogni esperimento, devi essere consapevole del suo ambito e del relativo impatto. È consigliabile eseguire la simulazione dell'errore in un ambiente non di produzione prima di eseguirla in un ambiente di produzione vero e proprio.

Gli esperimenti andrebbero eseguiti in produzione con un carico reale mediante distribuzioni canary che attivano l'implementazione di sistemi sperimentali e di controllo, laddove possibile. L'esecuzione degli esperimenti durante gli orari non di punta è altamente consigliata al fine di ridurre al massimo potenziali eventi negativi durante la prima esecuzione dell'esperimento negli ambienti di produzione. Inoltre, se l'utilizzo dell'effettivo traffico clienti costituisce un rischio eccessivo, puoi eseguire gli esperimenti utilizzando una sintesi del traffico nell'infrastruttura di produzione utilizzando implementazioni sperimentali e di controllo. Se l'utilizzo di un ambiente di produzione non è possibile, esegui gli esperimenti in ambienti di pre-produzione il più simili possibile agli effettivi ambienti di produzione.

Devi definire e monitorare i guardrail per essere sicuro che l'esperimento non abbia un impatto sul traffico di produzione o sugli altri sistemi che superi i limiti accettabili. Definisci condizioni di arresto per interrompere l'esperimento se viene raggiunta la soglia definita nella metrica del guardrail. In tali condizioni devono essere incluse le metriche relative allo stato stazionario del carico di lavoro e le metriche riferite ai componenti in cui inserisci l'errore. Un monitoraggio sintetico (definito anche canary utente) è una metrica che in genere deve essere inclusa come proxy utente. Le condizioni di arresto per AWS FIS sono supportate nel modello di esperimento, nella misura di un massimo di cinque condizioni di arresto per modello.

Uno dei principi dell'ingegneria del caos prevede la riduzione dell'ambito dell'esperimento e del relativo impatto.

Se da un lato deve essere prevista la possibilità di un determinato impatto negativo a breve termine, dall'altro il contenimento e la riduzione delle conseguenze negative degli esperimenti sono una responsabilità esclusiva dell'addetto all'ingegneria del caos.

Un metodo per verificare l'ambito e il potenziale impatto prevede l'esecuzione dell'esperimento dapprima in un ambiente non di produzione, la verifica che le soglie delle condizioni di arresto vengano attivate come previsto durante lo svolgimento di un esperimento e l'utilizzo effettivo delle misure di osservabilità finalizzate all'acquisizione di un'eccezione, anziché eseguire l'esperimento direttamente in produzione.

Durante l'esecuzione di esperimenti di iniezione di guasti, verifica che tutte le parti responsabili ne siano a conoscenza. Comunica ai team appropriati, ad esempio i team responsabili delle operazioni, dell'affidabilità dei servizi e del supporto clienti, quando verranno eseguiti gli esperimenti e l'impatto previsto. Metti a disposizione di questi team strumenti di comunicazione che consentano loro di informare i responsabili dell'esperimento di eventuali effetti avversi.

È necessario ripristinare lo stato originario del carico di lavoro e dei relativi sistemi sottostanti. La progettazione resiliente del carico di lavoro è spesso caratterizzata da funzionalità di riparazione automatica. Tuttavia, alcune progettazioni con difetti o alcuni esperimenti non riusciti possono compromettere in modo imprevisto lo stato del carico di lavoro. Entro la fine dell'esperimento dovrai essere consapevole di questa situazione e ripristinare il carico di lavoro e i sistemi. Con AWS FIS puoi impostare una configurazione di rollback, definita anche post-operazione, all'interno dei parametri operativi. Una post-operazione ripristina una destinazione allo stato in cui si trovava prima dell'esecuzione dell'operazione stessa. Indipendentemente dal fatto che vengano eseguite in modalità automatica, ad esempio utilizzando AWS FIS, o manuale, queste post-operazioni devono essere incluse in un playbook in cui vengono descritte le procedure di rilevamento e gestione degli errori.

d. Verifica l'ipotesi.

   Principles of Chaos Engineering fornisce le linee guida su come verificare lo stato stazionario del carico di lavoro.

   È necessario concentrarsi sull'output misurabile di un sistema e non sugli attributi interni del sistema. Le misurazioni di tale output in un breve periodo di tempo costituiscono un'attestazione dello stato stazionario del sistema. Il throughput del sistema nel suo complesso, le percentuali di errori e i percentili della latenza possono essere considerati metriche di interesse che rappresentano il comportamento di uno stato stazionario. Sulla base dei rilevamenti dei modelli di comportamento sistematico durante gli esperimenti, l'ingegneria del caos verifica che il sistema funzioni correttamente anziché tentare di convalidare il modo in cui funziona.

   Nei due esempi precedenti sono state incluse le metriche dello stato stazionario relative a un incremento inferiore allo 0,01% di errori (5xx) lato server e inferiore a un minuto di errori di lettura e scrittura del database.

   Gli errori 5xx rappresentano una buona metrica perché sono la conseguenza della modalità di errore che un client del carico di lavoro sperimenterà direttamente. La misurazione degli errori del database risulta valida come conseguenza diretta dell'errore, ma deve essere supportata da una misurazione diretta dell'impatto, ad esempio le richieste cliente non riuscite o gli errori restituiti a livello di client. Includi anche un monitoraggio sintetico, definito canary utente, in qualsiasi API o URI a cui il client del carico di lavoro ha accesso diretto.

e. Migliora la progettazione del carico di lavoro con un occhio di riguardo per la resilienza.

Se lo stato stazionario non è stato preservato, analizza in che modo puoi migliorare la progettazione del carico di lavoro per azzerare l'impatto dell'errore applicando le best practice illustrate nel pilastro dell'affidabilità di AWS Well-Architected. Puoi trovare ulteriori informazioni nella AWS Builder's Library, che offre, tra gli altri, articoli su come migliorare i controlli dell'integrità o impiegare nuovi tentativi con backoff nel codice dell'applicazione.

Dopo aver implementato queste modifiche, esegui di nuovo l'esperimento (rappresentato dalla linea punteggiata nel volano relativo all'ingegneria del caos) per determinare la relativa efficacia. Se nella fase di verifica risulta che l'ipotesi è vera, il carico di lavoro sarà in stato stazionario e il ciclo continuerà.

4. Esegui gli esperimenti con regolarità.

   Un esperimento di ingegneria del caos è un ciclo e gli esperimenti devono essere eseguiti regolarmente nell'ambito dell'ingegneria del caos. Se un carico di lavoro è conforme all'ipotesi dell'esperimento, l'esperimento deve essere automatizzato affinché venga eseguito continuamente come fase di regressione della pipeline CI/CD. Per ulteriori informazioni su come effettuare tale operazione, consulta questo blog su come eseguire esperimenti AWS FIS mediante AWS CodePipeline. Poi fare pratica con questo lab sugli esperimenti AWS FIS ricorrenti in una pipeline CI/CD.

   Gli esperimenti di iniezione di guasti fanno inoltre parte delle giornate di gioco (consulta REL12-BP05 Esecuzione regolare di GameDay). Le giornate di gioco simulano un errore o un evento per verificare sistemi, processi e risposte dei team. Lo scopo è di eseguire effettivamente le azioni che compirebbe il team come se si verificasse un evento eccezionale.

5. Acquisisci e archivia i risultati degli esperimenti.

   I risultati degli esperimenti di iniezione di guasti devono essere acquisiti e resi persistenti. Includi tutti i dati necessari, ad esempio orari, carico di lavoro e condizioni, in modo da essere in grado di analizzare i risultati e i trend in un secondo momento. I risultati potrebbero includere, ad esempio, screenshot dei pannelli di controllo, dump in formato CSV del database delle metriche oppure appunti scritti a mano relativi a eventi e osservazioni associati all'esperimento. Puoi inserire la creazione di log degli esperimenti mediante AWS FIS nel processo di acquisizione dei dati.

Risorse

Best practice correlate:

- REL08-BP03 Esecuzione di test di resilienza come parte integrante dell'implementazione
- REL13-BP03 Esecuzione di test sull'implementazione del disaster recovery per convalidare l'implementazione

Documenti correlati:

- Cos'è AWS Fault Injection Service?
- Cos'è AWS Resilience Hub?
- Principles of Chaos Engineering
- Chaos Engineering: Planning your first experiment
- Resilience Engineering: Learning to Embrace Failure
- Chaos Engineering stories
- Evitare il fallback nei sistemi distribuiti
- Distribuzione canary per gli esperimenti di ingegneria del caos

Video correlati:

- AWS re:Invent 2020: Testing resiliency using chaos engineering (ARC316)
- AWS re:Invent 2019: Improving resiliency with chaos engineering (DOP309-R1)
- AWS re:Invent 2019: Performing chaos engineering in a serverless world (CMY301)

Strumenti correlati:

- AWS Fault Injection Service
- Marketplace AWS: Gremlin Chaos Engineering Platform
- Chaos Toolkit
- Chaos Mesh
- Litmus

REL12-BP05 Esecuzione regolare di GameDay

Conduci GameDay per esercitare regolarmente le tue procedure di risposta agli eventi e alle compromissioni che incidono sul carico di lavoro. Coinvolgi gli stessi team responsabili della gestione

degli scenari di produzione. Queste esercitazioni aiutano ad applicare le misure per prevenire l'impatto sugli utenti causato da eventi di produzione. Esercitando le procedure di risposta in condizioni realistiche, puoi identificare e risolvere eventuali lacune o punti deboli prima che si verifichi un evento reale.

I GameDay simulano eventi in ambienti simili a quelli di produzione per testare sistemi, processi e risposte dei team. Lo scopo è quello di eseguire le stesse azioni che verrebbero eseguite dal team se l'evento si verificasse realmente. Questi esercizi aiutano a capire dove è possibile apportare miglioramenti e contribuire a sviluppare l'esperienza organizzativa nel gestire eventi e compromissioni. Questi dovrebbero essere svolti regolarmente, in modo che il team sappia costruire abitudini radicate su come rispondere.

I GameDay preparano i team a gestire gli eventi di produzione con maggiore sicurezza. I team ben allenati sono più in grado di individuare e rispondere rapidamente ai vari scenari. Ciò si traduce in un significativo miglioramento della postura di prontezza e resilienza.

Risultato desiderato: conduci GameDay sulla resilienza in modo coerente e programmato. Questi GameDay sono visti come una parte normale e attesa dell'attività. La tua organizzazione ha costruito una cultura di preparazione e quando si verificano problemi di produzione, i team sono ben preparati a rispondere efficacemente, a risolvere i problemi in modo efficiente e a mitigare l'impatto sui clienti.

Anti-pattern comuni:

- Documenti le procedure, ma non le metti mai in pratica.
- Negli esercizi di prova escludi i responsabili delle decisioni aziendali.
- Organizzi un GameDay, ma non informi tutte le parti interessate.
- Ti concentri esclusivamente sugli errori tecnici, ma non coinvolgi le parti interessate aziendali.
- Non incorpori le lezioni apprese nei GameDay nei processi di recupero.
- Incolpi i team per errori o bug.

Vantaggi dell'adozione di questa best practice:

- Migliorare le capacità di risposta: nei GameDay, i team si esercitano a svolgere i propri compiti e a testare i meccanismi di comunicazione durante gli eventi simulati, creando una risposta più coordinata ed efficiente nelle situazioni di produzione.
- Identifica e risolvi le dipendenze: gli ambienti complessi spesso comportano dipendenze intricate tra vari sistemi, servizi e componenti. I GameDay possono aiutare a identificare e

risolvere queste dipendenze e a verificare che i sistemi e i servizi critici siano adeguatamente coperti dalle procedure del runbook e che possano essere aumentati verticalmente o ripristinati tempestivamente.

- Promuovere una cultura della resilienza: i GameDay possono aiutare a coltivare una mentalità di resilienza all'interno di un'organizzazione. Quando si coinvolgono team interfunzionali e parti interessate, questi esercizi promuovono la consapevolezza, la collaborazione e la comprensione condivisa dell'importanza della resilienza in tutta l'organizzazione.
- Miglioramento e adattamento continui: GameDay regolari aiutano a valutare e adattare continuamente le strategie di resilienza, in modo da mantenerle pertinenti ed efficaci di fronte a circostanze mutevoli.
- Aumentare la fiducia nel sistema: GameDay riusciti possono aiutare a creare fiducia nella capacità del sistema di resistere e riprendersi da interruzioni.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Una volta progettate e implementate le misure di resilienza necessarie, conduci un GameDay per verificare che tutto funzioni come previsto in produzione. Un GameDay, soprattutto il primo, deve coinvolgere tutti i membri del team. Tutte le parti interessate e i partecipanti devono essere informati in anticipo su data, ora e scenari simulati.

Durante il GameDay, i team coinvolti simulano vari eventi e potenziali scenari secondo le procedure prescritte. I partecipanti monitorano e valutano attentamente l'impatto di questi eventi simulati. Se il sistema funziona come previsto, i meccanismi automatici di rilevamento, dimensionamento e autoriparazione devono attivarsi e generare un impatto minimo o nullo sugli utenti. Se il team rileva un impatto negativo, esegue il rollback del test e corregge i problemi identificati, sia con mezzi automatici sia con interventi manuali documentati nei runbook applicabili.

Per migliorare continuamente la resilienza, è fondamentale documentare e incorporare le lezioni apprese. Questo processo è un ciclo di feedback che acquisisce sistematicamente le intuizioni dei GameDay e le utilizza per migliorare i sistemi, i processi e le capacità del team.

Per aiutare a riprodurre scenari reali in cui i componenti o i servizi del sistema possono generare errori imprevisti, si consiglia di iniettare errori simulati come esercizio del GameDay. I team possono testare la resilienza e la tolleranza agli errori dei loro sistemi e simulare i processi di risposta e di ripristino agli incidenti in un ambiente controllato.

In AWS, i GameDay possono essere realizzati con repliche dell'ambiente di produzione utilizzando infrastructure as code. Questo processo consente di eseguire i test in un ambiente sicuro e molto simile a quello di produzione. Prendi in considerazione il servizio AWS Fault Injection Service per creare diversi scenari di errore. Utilizza servizi come Amazon CloudWatch e AWS X-Ray per monitorare il comportamento del sistema durante i GameDay. Utilizza AWS Systems Manager per gestire ed eseguire i playbook e utilizza AWS Step Functions per orchestrare i flussi di lavoro ricorrenti del GameDay.

Passaggi dell'implementazione

- Stabilisci un programma per i GameDay: sviluppa un programma strutturato che definisce la frequenza, la portata e gli obiettivi dei GameDay. Coinvolgi le principali parti interessate e gli esperti in materia nella pianificazione e nello svolgimento di questi esercizi.
- Prepara il GameDay:
  1. Identifica i servizi chiave critici per l'azienda che sono al centro del GameDay. Cataloga e mappa le persone, i processi e le tecnologie che supportano tali servizi.
  2. Stabilisci il programma del GameDay e prepara i team coinvolti a partecipare all'evento. Prepara i servizi di automazione per simulare gli scenari pianificati ed esegui i processi di ripristino appropriati. I servizi AWS come AWS Fault Injection Service, AWS Step Functions e AWS Systems Manager possono aiutarti ad automatizzare vari aspetti dei GameDay, come l'iniezione di errori e l'avvio di azioni di ripristino.
- Esegui la simulazione: nel GameDay, esegui lo scenario pianificato. Osserva e documenta come le persone, i processi e le tecnologie reagiscono all'evento simulato.
- Conduci revisioni post-esercizio: dopo il GameDay, conduci una sessione retrospettiva per esaminare le lezioni apprese. Identifica le aree di miglioramento e le azioni necessarie per migliorare la resilienza operativa. Documenta gli esiti e tieni traccia delle eventuali modifiche necessarie per migliorare le strategie di resilienza e la preparazione al completamento.

Risorse

Best practice correlate:

- REL12-BP01 Utilizzo dei playbook per analizzare gli errori
- REL12-BP04 Test della resilienza tramite l'utilizzo dell'ingegneria del caos
- OPS04-BP01 Identificazione degli indicatori chiave di prestazione
- OPS07-BP03 Utilizzo di runbook per eseguire le procedure

- OPS10-BP01 Utilizzo di un processo per la gestione di eventi, incidenti e problemi

Documenti correlati:

- Che cos'è AWS GameDay?

Video correlati:

- AWS re:Invent 2023 - Practice like you play: How Amazon scales resilience to new heights

Esempi correlati:

- AWS Workshop - Navigate the storm: Unleashing controlled chaos for resilient systems
- Build Your Own Game Day to Support Operational Resilience

REL 13. Come si pianifica il disaster recovery?

Avere backup e componenti del carico di lavoro ridondanti in loco è l'inizio della strategia di disaster recovery. RTO ed RPO sono gli obiettivi per il ripristino del carico di lavoro. Imposta questi valori in base alle esigenze aziendali. Implementa una strategia per raggiungere questi obiettivi, prendendo in considerazione le posizioni e la funzione delle risorse e dei dati del carico di lavoro. La probabilità di interruzione e il costo del ripristino sono fattori chiave che aiutano a comunicare il valore aziendale che può avere il disaster recovery per un carico di lavoro.

Best practice

- REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati
- REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino
- REL13-BP03 Esecuzione di test sull'implementazione del disaster recovery per convalidare l'implementazione
- REL13-BP04 Gestione della deviazione di configurazione nel sito o nella regione del disaster recovery
- REL13-BP05 Automatizzazione del ripristino

REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati

Guasti ed errori possono avere un impatto sull'attività in diversi modi. In primo luogo, possono causare l'interruzione del servizio (tempo di inattività). In secondo luogo, possono causare la perdita, l'incoerenza o il mancato aggiornamento dei dati. Per guidare le modalità di risposta e recupero dagli errori, definisci un Obiettivo del tempo di ripristino (RTO) e un Obiettivo del punto di ripristino (RPO) per ogni carico di lavoro. L'Obiettivo del tempo di ripristino (RTO) è il ritardo massimo accettabile tra l'interruzione del servizio e il ripristino del servizio. L'Obiettivo del punto di ripristino (RPO) è il tempo massimo accettabile dopo l'ultimo punto di ripristino dei dati.

Risultato desiderato: ogni carico di lavoro ha un RTO e un RPO designati in base a considerazioni tecniche e all'impatto aziendale.

Anti-pattern comuni:

- Non hai designato gli obiettivi di ripristino.
- Selezioni obiettivi di ripristino arbitrari.
- Selezioni obiettivi di ripristino troppo blandi e che non soddisfano gli obiettivi aziendali.
- Non hai valutato l'impatto del tempo di inattività e della perdita di dati.
- Scegli obiettivi di ripristino non realistici, come il tempo zero di ripristino o nessuna perdita di dati, che potrebbero non essere raggiungibili per la configurazione del carico di lavoro.
- Selezioni obiettivi di ripristino più severi rispetto agli obiettivi aziendali reali. Questo costringe a implementazioni di ripristino più costose e complicate rispetto alle esigenze del carico di lavoro.
- Selezioni obiettivi di ripristino incompatibili con quelli di un carico di lavoro dipendente.
- Non tieni conto dei requisiti normativi e di conformità.

Vantaggi dell'adozione di questa best practice: quando definisci gli RTO e gli RPO per i carichi di lavoro, stabilisci obiettivi chiari e misurabili per il ripristino in base alle esigenze aziendali. Una volta fissati questi obiettivi, puoi creare piani di disaster recovery (DR) su misura per raggiungerli.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Costruisci una matrice o un foglio di lavoro per guidare la pianificazione del disaster recovery. Nella matrice, crea diverse categorie o livelli di carico di lavoro in base al loro impatto sull'azienda (ad

esempio, critico, alto, medio e basso) e i relativi RTO e RPO da raggiungere per ciascuno di essi. La matrice seguente fornisce un possibile esempio (nota che i valori RTO e RPO possono differire) da seguire:

Matrice di ripristino di emergenza						
		Obiettivo del punto di ripristino				
		meno di 1 minuto	meno di 1 ora	meno di 6 ore	meno di 1 giorno	Più di 1 giorno
Obiettivo del tempo di ripristino	meno di 10 minuti	Critica	Critica	Alta	Medio	Medio
	meno di 2 ore	Critica	Alta	Medio	Medio	Bassa
	meno di 8 ore	Alta	Medio	Medio	Bassa	Bassa
	meno di 24 ore	Medio	Medio	Bassa	Bassa	Bassa
	Più di 24 ore	Medio	Bassa	Bassa	Bassa	Bassa

Esempio di matrice di disaster recovery

Per ogni carico di lavoro, devi analizzare e comprendere l'impatto sull'azienda del tempo di inattività e della perdita di dati. L'impatto cresce tipicamente con il tempo di inattività e la perdita di dati, ma la forma dell'impatto può variare in base al tipo di carico di lavoro. Ad esempio, un tempo di inattività fino a un'ora potrebbe avere un impatto ridotto, ma in seguito l'impatto potrebbe intensificarsi rapidamente. L'impatto può assumere diverse forme, tra cui l'impatto finanziario (come la perdita di fatturato), l'impatto a livello di reputazione (tra cui la perdita di fiducia dei clienti), l'impatto operativo (come il mancato pagamento degli stipendi o la diminuzione della produttività) e il rischio normativo. Una volta completato, assegna il carico di lavoro al livello appropriato.

Considera le seguenti domande quando analizzi l'impatto del guasto o dell'errore:

1. Qual è il tempo massimo di indisponibilità del carico di lavoro prima che si verifichi un impatto inaccettabile sull'azienda?
2. Qual è l'intensità e il tipo di impatto che l'azienda subirà a causa di un'interruzione del carico di lavoro? Prendi in considerazione tutti i tipi di impatto, compresi quelli finanziari, a livello di reputazione, operativi e normativi.
3. Qual è la quantità massima di dati che può essere persa o non recuperata prima che si verifichi un impatto inaccettabile sull'azienda?
4. I dati persi possono essere ricreati da altre fonti (note anche come dati derivati)? In tal caso, considera anche gli RPO di tutti i dati di origine utilizzati per ricreare i dati del carico di lavoro.

5. Quali sono gli obiettivi di ripristino e le aspettative di disponibilità dei carichi di lavoro da cui questo dipende (a valle)? Gli obiettivi del carico di lavoro devono essere raggiungibili in base alle capacità di ripristino delle relative dipendenze a valle. Valuta possibili soluzioni alternative o mitigazioni delle dipendenze downstream che possono migliorare la capacità di ripristino di questo carico di lavoro.
6. Quali sono gli obiettivi di ripristino e le aspettative di disponibilità dei carichi di lavoro che dipendono da questo (upstream)? Gli obiettivi del carico di lavoro upstream possono richiedere che questo carico di lavoro disponga di capacità di ripristino più rigorose di quanto non sembri a prima vista.
7. Esistono obiettivi di recupero diversi in base al tipo di incidente? Ad esempio, si possono avere RTO e RPO diversi a seconda che l'incidente riguardi una zona di disponibilità o un'intera Regione.
8. Gli obiettivi di ripristino cambiano durante determinati eventi o periodi dell'anno? Ad esempio, si possono avere RTO e RPO diversi in base alle stagioni dello shopping, agli eventi sportivi, alle vendite speciali e al lancio di nuovi prodotti.
9. In che modo gli obiettivi di ripristino si allineano con la strategia di disaster recovery aziendale e organizzativa?
10. Ci sono implicazioni legali o contrattuali da considerare? Ad esempio, hai l'obbligo per contratto di fornire un servizio con un determinato RTO o RPO? In quali sanzioni potresti incorrere in caso di inadempienza?
11. Devi mantenere l'integrità dei dati per soddisfare i requisiti normativi o di conformità?

Il seguente foglio di lavoro può aiutarti a valutare ogni carico di lavoro. Puoi modificare questo foglio di lavoro per adattarlo alle tue esigenze specifiche, ad esempio aggiungendo altre domande.

Passo 2: domande principali	Si applica al carico di lavoro?	RTO del carico di lavoro	RPO del carico di lavoro	RTO rettif.	RPO rettif.	Istruzioni
[1] tempo massimo di inattività del carico di lavoro						misurato in tempo dall'inizio del malfunzionamento al ripristino
[2] quantità massima di dati che possono essere persi						misurato in tempo trascorso dall'ultimo set di dati integro ripristinabile
[3a] dipendenze a monte						inserire gli obiettivi di recupero a monte più rigorosi
[3b] riconciliazione delle dipendenze a valle						inserire gli obiettivi di recupero a valle meno rigorosi
[3a] riconciliazione delle dipendenze a monte						Se il valore a monte è inferiore ai valori attuali e il valore a valle è superiore,
[3b] riconciliazione delle dipendenze a valle						operare sulle dipendenze per riconciliare i valori e inserirli qui.
[3] dipendenze						ridurre i valori per soddisfare le dipendenze a monte o alzarli in base alle capacità delle dipendenza a valle
Passo 2: domande aggiuntive						Indicare se la domanda è pertinente. Saltarla in caso affermativo
RTO/RPO di base						Riportare qui i valori di RTO e RPO sopra indicati
[4] tipo di malfunzionamento	[ ]S / [ ]N					Inserire gli obiettivi di recupero per i tipi di evento con i requisiti più rigorosi
[5] obiettivi specifici basati sul tempo	[ ]S / [ ]N					Inserire gli obiettivi di recupero per i tempi con i requisiti più rigorosi
[6] clienti che sperimentano il disservizio	[ ]S / [ ]N					Tracciare un grafico dei clienti che sperimentano il disservizio in funzione del tempo di inattività o dei dati persi. Utilizzare tale grafico per inserire i valori massimi di RTO e RPO ammissibili in base all'impatto sui clienti
[7] impatto reputazionale	[ ]S / [ ]N					Lavorare in modo congiunto con l'azienda per determinare i massimi valori di RTO e RPO in base all'impatto sulla reputazione
[8] impatto operativo	[ ]S / [ ]N					Inserire i valori massimi di RTO e RPO sulla base dell'impatto operativo
[9] allineamento aziendale	[ ]S / [ ]N					Inserire i valori massimi di RTO e RPO per i carichi di lavoro di questo tipo in base ai requisiti LOB e organizzativi
[10] obblighi contrattuali	[ ]S / [ ]N					Inserire i valori massimi di RTO e RPO sulla base degli obblighi contrattuali
[11] conformità normativa	[ ]S / [ ]N					Inserire i valori massimi di RTO e RPO sulla base delle norme di conformità applicabili
obiettivo sulla base delle domande aggiuntive						Selezionare il valore minimo (valore più rigoroso) dalle domande 4-11 e inserirlo qui
obiettivo rettificato						Se non è possibile raggiungere gli obiettivi indicati nella riga precedente, collaborare con le parti interessate per allentare i vincoli e inserire un nuovo minimo qui.
RTO/RPO rettificato						Inserire il valore inferiore tra RPO/RTO di base e valore obiettivo rettificato
Passo 3						
Mappatura su categorie o livelli predefiniti						Regolare entrambi i valori verso il basso (requisito più rigoroso) per allinearsi al livello più vicino definito

Foglio di lavoro

Passaggi dell'implementazione

1. Identifica le parti interessate aziendali e i team tecnici responsabili di ciascun carico di lavoro e collabora con loro.
2. Crea categorie o livelli di criticità per l'impatto del carico di lavoro nell'organizzazione. Le categorie di esempio sono: critico, alto, medio e basso. Per ogni categoria, scegli un RTO e un RPO che riflettano gli obiettivi e i requisiti aziendali.
3. Assegna a ciascun carico di lavoro una delle categorie di impatto create nel passaggio precedente. Per decidere in che modo un carico di lavoro rientra in una categoria, considera l'importanza del carico di lavoro per l'azienda e l'impatto di un'interruzione o di una perdita di dati e utilizza le domande di cui sopra come guida. Ne conseguono un RTO e un RPO per ogni carico di lavoro.
4. Considera l'RTO e l'RPO per ogni carico di lavoro determinato nel passaggio precedente. Coinvolgi i team aziendali e tecnici del carico di lavoro per determinare se gli obiettivi devono essere modificati. Ad esempio, le parti interessate aziendali potrebbero stabilire che siano necessari obiettivi più severi. In alternativa, i team di tecnici potrebbero decidere di modificare gli obiettivi per renderli raggiungibili con le risorse disponibili e i vincoli tecnologici.

Risorse

Best practice correlate:

- REL09-BP04 Ripristino periodico dei dati per verificare l'integrità e i processi di backup
- REL12-BP01 Utilizzo dei playbook per analizzare gli errori
- REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino
- REL13-BP03 Esecuzione di test sull'implementazione del disaster recovery per convalidare l'implementazione

Documenti correlati:

- AWS Architecture Blog: serie sul disaster recovery
- Ripristino di emergenza dei carichi di lavoro su AWS: ripristino nel cloud (whitepaper di AWS)
- Managing resiliency policies with AWS Resilience Hub
- Partner APN: partner che possono assistere con il disaster recovery
- Marketplace AWS: prodotti utilizzabili per il disaster recovery

Video correlati:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications
- Disaster Recovery of Workloads on AWS

REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino

Definisci una strategia di disaster recovery che soddisfi gli obiettivi di ripristino del carico di lavoro. Scegli una strategia, ad esempio backup e ripristino, standby (attivo/passivo) o attivo/attivo.

Risultato desiderato: per ciascun carico di lavoro esiste una strategia di disaster recovery definita e implementata che consente a quel carico di lavoro di raggiungere gli obiettivi di disaster recovery. Le strategie di disaster recovery tra carichi di lavoro utilizzano modelli riutilizzabili (come strategie descritte in precedenza),

Anti-pattern comuni:

- Implementazione di procedure di ripristino incoerenti per carichi di lavoro con obiettivi di ripristino simili.

- Implementazione di una strategia di disaster recovery ad-hoc quando si verifica un disastro.
- Assenza di piani per il disaster recovery.
- Dipendenza dalle operazioni del piano di controllo (control-plane) durante il ripristino.

Vantaggi dell'adozione di questa best practice:

- L'utilizzo di strategie di ripristino definite consente di utilizzare strumenti e procedure di test comuni.
- L'uso di strategie di ripristino definite permette la condivisione delle informazioni tra team e l'implementazione del disaster recovery nei carichi dl lavoro di loro proprietà.

Livello di rischio associato se questa best practice non fosse adottata: elevato. Senza una strategia di disaster recovery pianificata, implementata e testata, è poco probabile riuscire a raggiungere gli obiettivi di ripristino in caso di emergenze.

Guida all'implementazione

Una strategia di disaster recovery si basa sulla capacità di creare il tuo carico di lavoro in un sito di ripristino se la tua sede principale non è disponibile per eseguire il carico di lavoro. Gli obiettivi di ripristino più comuni sono RTO e RPO, come discusso in REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati.

Una strategia di disaster recovery (DR) su più zone di disponibilità (AZ) all'interno di un singolo Regione AWS può offrire la mitigazione rispetto a emergenze come incendi, alluvioni e interruzioni gravi dell'energia. Se è un requisito implementare una protezione rispetto a un evento improbabile che impedisca al tuo carico di lavoro di poter essere eseguito in un determinato Regione AWS, puoi usare una disaster recovery di emergenza basata su più regioni.

Quando pianifichi una strategia di disaster recovery su più regioni, devi scegliere una delle seguenti strategie. Sono elencati in ordine crescente di costo e complessità e in ordine decrescente di RTO e RPO. La regione di ripristino indica una Regione AWS diversa da quella principale utilizzata per il carico di lavoro.

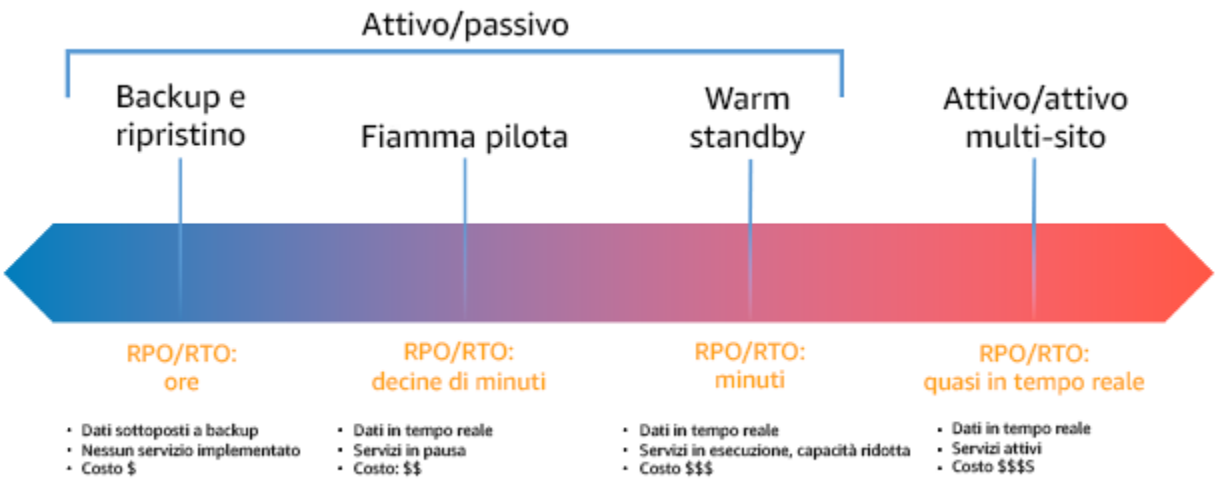


Figura 17: strategie di disaster recovery (DR)

- Backup e ripristino (RPO in ore, RTO in 24 ore o meno): esegui il backup dei dati e delle applicazioni nella regione di recupero. Adottando backup continui o automatizzati otterrai un ripristino point-in-time (PITR) che può ridurre il valore dell'RPO fino a raggiungere in alcuni casi 5 minuti. Nel caso in cui si verifichi un disastro, distribuirai l'infrastruttura (usando il modello Infrastructure as code per ridurre l'RTO), implementerai il codice e ripristinerai i dati del backup dopo un disastro nella regione di ripristino.
- Pilot light (RPO in minuti, RTO in decine di minuti): fornisci una copia dell'infrastruttura del carico di lavoro di base nella regione di ripristino. Replica i dati nella regione di ripristino e crea un backup in essa. Le risorse necessarie per supportare la replica dei dati e il backup, come database e archiviazione di oggetti, sono sempre attive. Altri elementi come i server applicativi o il calcolo serverless non vengono distribuiti, ma possono essere creati quando necessari con la configurazione e il codice applicativo richiesti.
- Warm standby (RPO in secondi, RTO in minuti): mantieni sempre una versione ridotta del carico di lavoro completamente funzionante in esecuzione nella regione di ripristino. I sistemi business critical sono completamente duplicati e sono sempre accesi, ma con un parco istanze ridotto verticalmente. I dati vengono replicati e si trovano nella regione di recupero. Al momento del ripristino, il sistema viene fatto aumentare verticalmente rapidamente per gestire il carico di produzione. Più si aumenterà verticalmente nella strategia di Warm Standby, più bassi saranno l'RTO e la dipendenza del piano di controllo (control-plane). Quando il dimensionamento è completo, si parla di standby a caldo.

- Attivo/attivo multi-regione (multisito) (RPO vicino a zero, RTO uguale potenzialmente a zero): il carico di lavoro viene implementato in più Regioni AWS e serve attivamente il traffico da esse proveniente. Questa strategia comporta la sincronizzazione dei dati tra le regioni. È necessario evitare o gestire possibili conflitti causati da scritture sullo stesso record in due diverse repliche regionali, un'attività che potrebbe rivelarsi complessa. La replica dei dati è utile per la sincronizzazione dei dati e ti proteggerà da alcuni tipi di disastri, ma non dalla corruzione o dalla distruzione dei dati, a meno che la tua soluzione non includa opzioni per il ripristino point-in-time.

> **Note**
>
> La differenza tra Pilot Light e Warm Standby può talvolta essere difficile da comprendere. Entrambe prevedono un ambiente nella tua regione di ripristino con copie degli asset della tua regione principale. La differenza è che la strategia Pilot Light non può elaborare le richieste senza aver prima intrapreso altre azioni, mentre Warm Standby può gestire immediatamente il traffico (a livelli ridotti di capacità). La strategia Pilot Light richiede l'attivazione dei server, possibilmente l'implementazione di un'infrastruttura aggiuntiva (non principale) e l'aumentare verticalmente, mentre Warm Standby richiede solo l'aumentare verticalmente (tutto è già stato implementato ed è in esecuzione). Scegli tra queste opzioni in base alle tue esigenze di RTO e RPO.
> Quando i costi sono un motivo di preoccupazione e vuoi realizzare obiettivi RPO ed RTO simili a quelli definiti nella strategia di Warm Standby, puoi prendere in considerazione soluzioni cloud-native (native del cloud), come AWS Elastic Disaster Recovery, che adotta l'approccio Pilot Light e offre obiettivi RPO ed RTO migliori.

Passaggi dell'implementazione

1. Definisci una strategia di disaster recovery in linea con i requisiti di ripristino di questo carico di lavoro.

   La scelta di una strategia di disaster recovery è un compromesso tra la riduzione dei tempi di inattività e della perdita di dati (RTO ed RPO) e i costi e la complessità di implementazione della strategia. Dovresti evitare di implementare una strategia che sia più severa del necessario, in quanto questo comporterebbe costi aggiuntivi.

Ad esempio, nel diagramma seguente, l'azienda ha stabilito l'RTO massimo concesso e il limite di spesa per la strategia di ripristino del servizio. Considerati gli obiettivi dell'azienda, le strategie di disaster recovery Pilot Light o di Warm Standby soddisfano sia l'RTO sia i criteri per i costi.

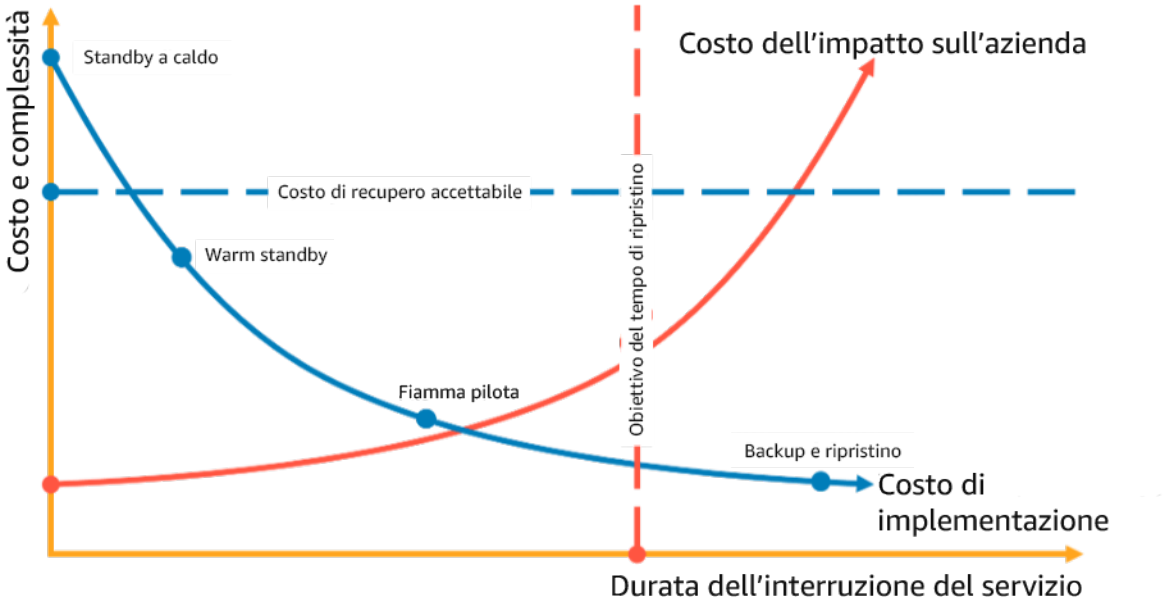


Figura 18: scegliere una strategia di disaster recovery in base all'RTO e ai costi

Per ulteriori informazioni, consulta Piano di continuità aziendale.

2. Esamina i modelli con cui la strategia di disaster recovery selezionata può essere implementata.

   Questo passaggio consiste nel capire come implementare la strategia selezionata. Le strategie vengono spiegate con Regioni AWS come siti principali e di ripristino. Tuttavia, puoi anche decidere di utilizzare le zone di disponibilità in una singola regione come strategia di disaster recovery, utilizzando aspetti di più strategie.

   Nei passaggi seguenti puoi applicare la strategia al carico di lavoro specifico.

   Backup e ripristino

   Backup ripristino è la strategia meno complessa da implementare, ma richiederà più tempo e impegno per ripristinare il carico di lavoro, generando così valori RTO e RPO più elevati. È buona pratica creare sempre backup dei dati e copiarli in un altro sito (ad esempio, un'altra Regione AWS).

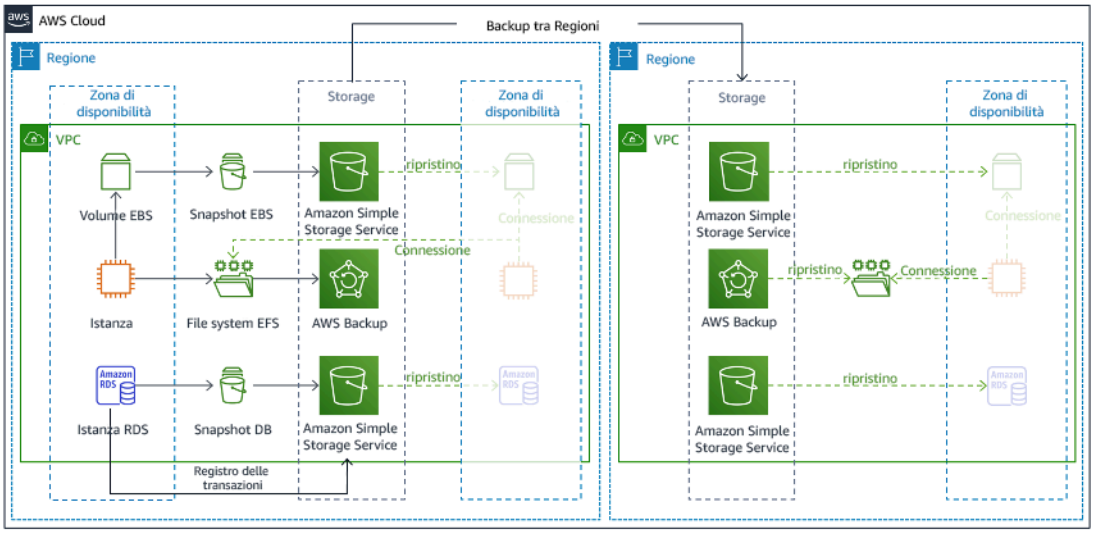


Figura 19: architettura di backup e ripristino

Per ulteriori dettagli su questa strategia, consulta [Disaster Recovery (DR) Architecture on AWS, Part II: Backup and Restore with Rapid Recovery].

Pilot light

Con l'approccio pilot light, replichi i dati dalla tua regione principale alla regione di ripristino. Le risorse di base utilizzate per l'infrastruttura del carico di lavoro vengono implementate nella regione di ripristino; tuttavia sono comunque necessarie risorse aggiuntive ed eventuali dipendenze per rendere funzionale questo stack. Ad esempio, nella Figura 20 non viene implementata alcuna risorsa di calcolo.

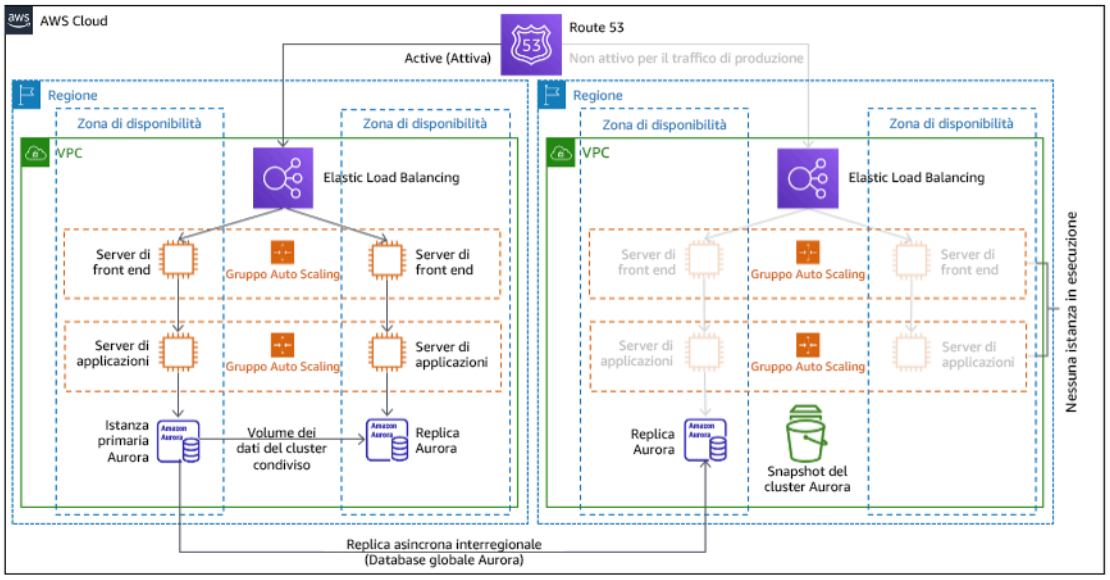


Figura 20: architettura pilot light

Per ulteriori informazioni su questa strategia, consulta [Disaster Recovery (DR) Architecture on AWS, Part III: Pilot Light and Warm Standby].

Warm standby

L'approccio warm standby implica la verifica della presenza di una copia ridotta verticalmente, ma comunque funzionale, dell'ambiente di produzione in un'altra regione. Questo approccio estende il concetto di Pilot Light e diminuisce il tempo di ripristino, poiché il carico di lavoro è sempre attivo in un'altra regione. Se la regione di ripristino ha raggiunto il massimo della capacità, allora viene definita come standby a caldo.

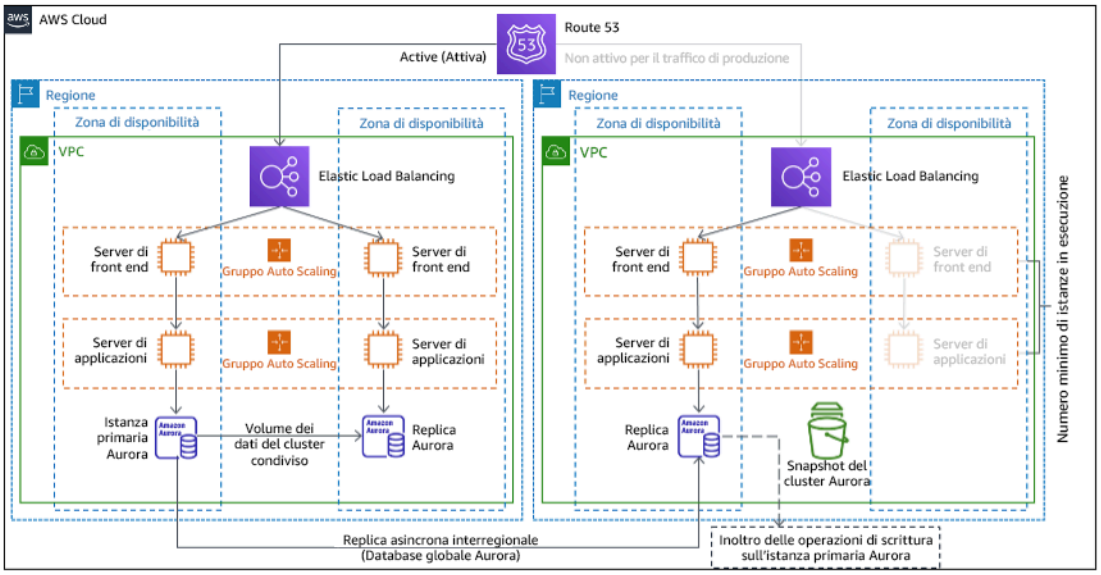


Figura 21: architettura warm standby

Se si utilizza Warm Standby o Pilot Light è necessario aumentare verticalmente le risorse nella regione di ripristino. Per verificare che sia disponibile capacità sufficiente quando necessario, valuta l'eventuale utilizzo delle prenotazioni della capacità per le istanze EC2. In caso di utilizzo di AWS Lambda, la concorrenza allocata può fornire ambienti di runtime pronti a rispondere immediatamente alle invocazioni della funzione.

Per ulteriori informazioni su questa strategia, consulta Disaster Recovery (DR) Architecture on AWS, Part III: Pilot Light and Warm Standby.

Attivo/attivo multi-sito

Puoi eseguire il carico di lavoro simultaneamente in più regioni come parte di una strategia attivo/attivo multi-sito. La strategia attivo/attivo multi-sito serve il traffico da tutte le regioni in cui è distribuita. I clienti possono selezionare questa strategia per motivi diversi dal disaster recovery. Può essere utilizzata per aumentare la disponibilità o nella distribuzione di un carico di lavoro a un pubblico globale (per posizionare l'endpoint più vicino agli utenti e/o per distribuire stack localizzati al pubblico di quella regione). Come strategia di disaster recovery, se il carico di lavoro non può essere supportato in una delle Regioni AWS in cui viene implementato, la regione viene evacuata e vengono usate le regioni rimanenti per garantire la disponibilità. La strategia attivo/attivo multi-sito è la strategia di ripristino operativamente più complessa e dovrebbe essere selezionata solo quando lo richiedono i requisiti aziendali.

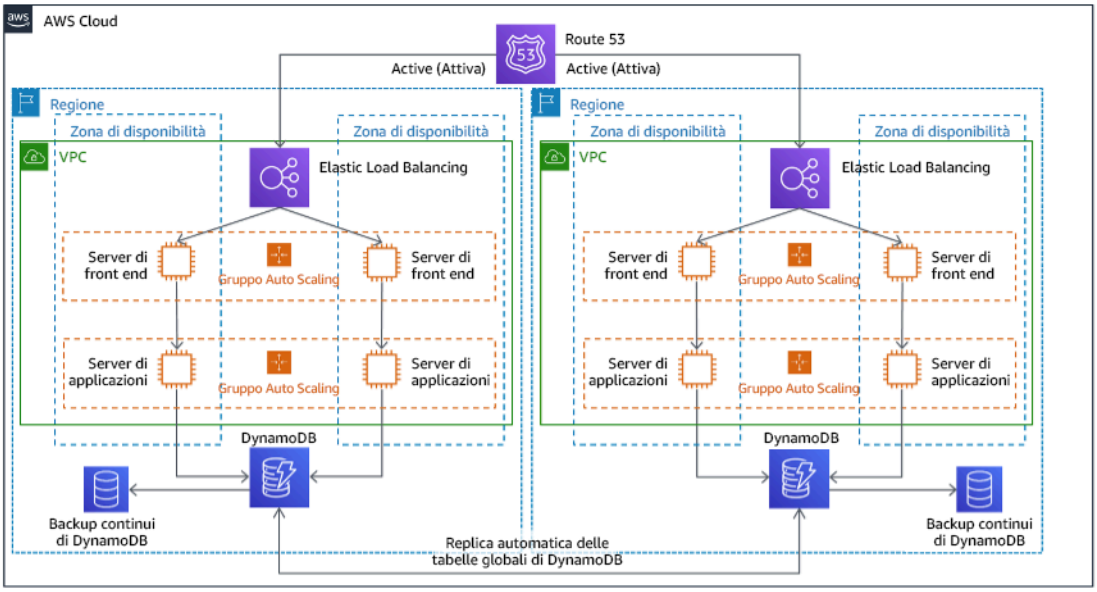


Figura 22: architettura attivo/attivo multi-sito

Per ulteriori informazioni su questa strategia, consulta [Disaster Recovery (DR) Architecture on AWS, Part IV: Multi-site Active/Active].

AWS Elastic Disaster Recovery

Se stai prendendo in considerazione la strategia pilot light o warm standby per il disaster recovery, AWS Elastic Disaster Recovery potrebbe fornire un approccio alternativo con maggiori vantaggi. Elastic Disaster Recovery può offrire obiettivi RPO e RTO simili alla strategia warm standby, ma con l'approccio a basso costo della strategia pilot light. Elastic Disaster Recovery replica i dati dalla regione primaria a quella di ripristino, usando una protezione continua dei dati per conseguire un RPO misurato in secondi e un RTO misurabile in minuti. Solo le risorse necessarie per replicare i dati vengono implementate nella regione di ripristino, mantenendo i costi ridotti come nella strategia Pilot Light. Quando usi Elastic Disaster Recovery, il servizio coordina e orchestra il ripristino delle risorse di calcolo quando viene avviato come parte di un failover o di un'esercitazione.

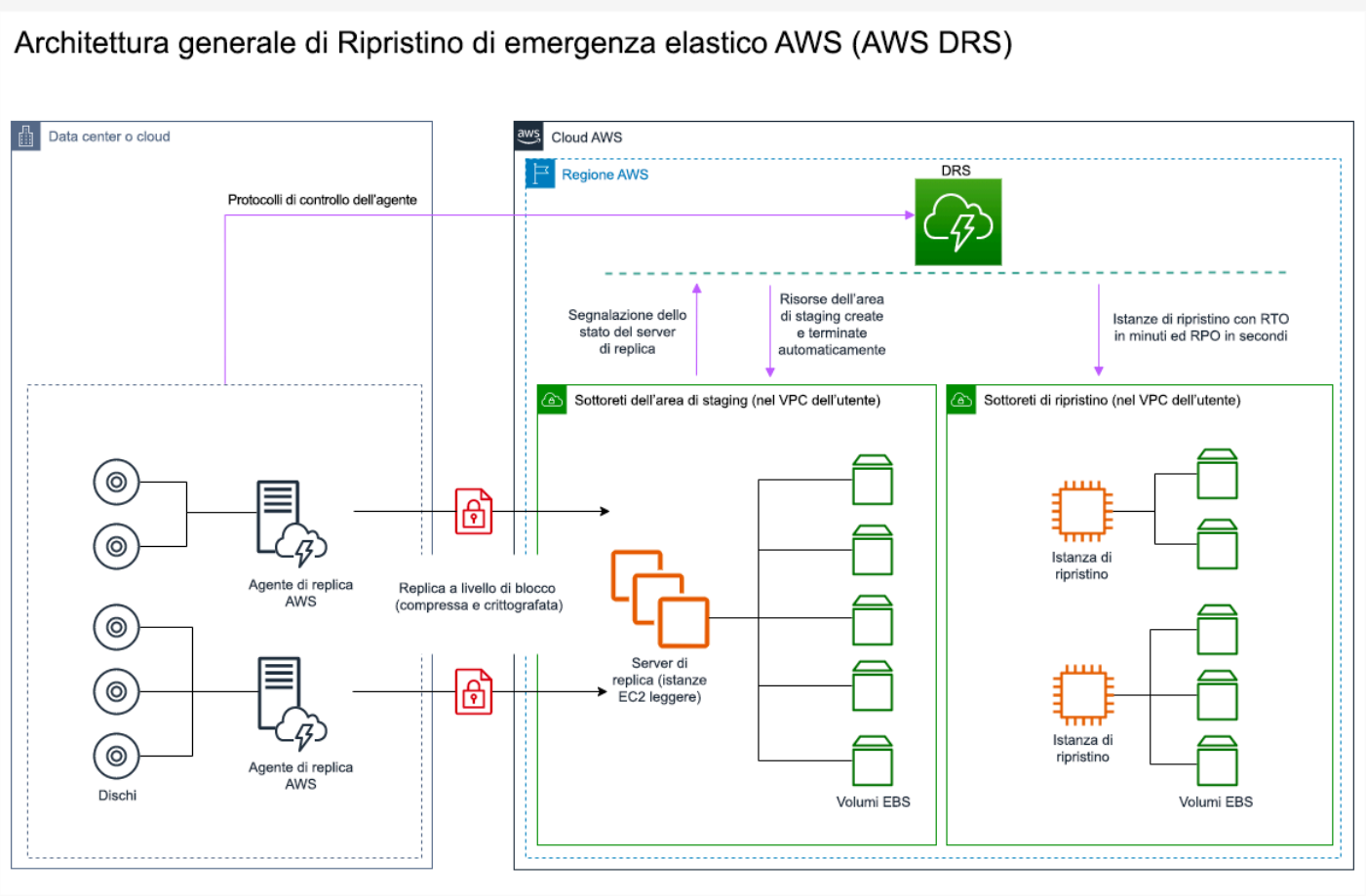


Figura 23: architettura AWS Elastic Disaster Recovery

Procedure aggiuntive per la protezione dei dati

Con tutte le strategie devi anche mitigare un disastro relativo ai dati. La replica continua dei dati ti proteggerà da alcuni tipi di disastri, ma non dalla corruzione o dalla distruzione dei dati, a meno che la tua soluzione non includa opzioni per il ripristino point-in-time o il controllo delle versioni dei dati archiviati. Devi anche creare un backup dei dati replicati nel sito di ripristino per creare backup point-in-time in aggiunta alle repliche.

Utilizzo di più zone di disponibilità all'interno di una singola Regione AWS

Quando si usano più zone di disponibilità all'interno di un'unica regione, l'implementazione della strategia di disaster recovery usa più elementi delle strategie precedenti. Devi innanzitutto creare un'architettura con disponibilità elevata usando più zone di disponibilità, come mostrato nella Figura 23. Questa architettura utilizza un approccio attivo/attivo multisito, in quanto le istanze Amazon EC2 ed Elastic Load Balancer dispongono di risorse implementate in più zone

di disponibilità, che gestiscono attivamente le richieste. L'architettura dimostra inoltre che lo standby a caldo, in cui in caso di errore dell'istanza Amazon RDS primaria (o della stessa zona di disponibilità), l'istanza di standby viene promossa a primaria.

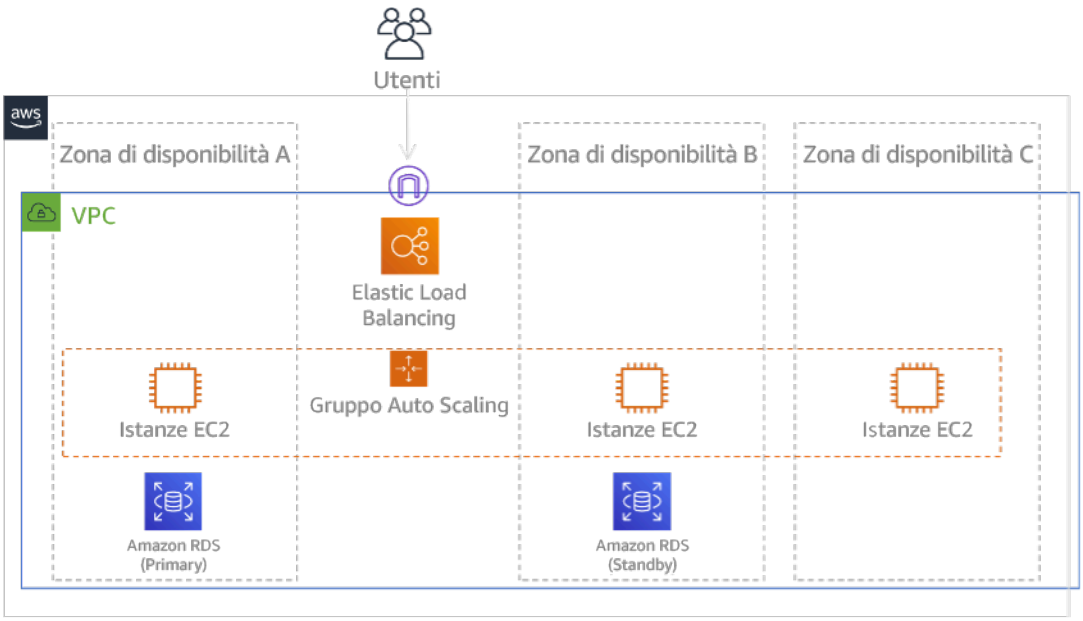


Figura 24: architettura con più zone di disponibilità

Oltre a questa architettura HA, devi aggiungere i backup di tutti i dati richiesti per eseguire il tuo carico di lavoro. Questo aspetto è di particolare importanza per i dati vincolati a una singola zona, come i volumi Amazon EBS o i cluster Amazon Redshift. In caso di errore di una zona di disponibilità, dovrai ripristinare i dati in un'altra zona di disponibilità. Laddove possibile, dovrai anche copiare i backup di dati su un'altra Regione AWS come forma di ulteriore protezione.

Un approccio alternativo meno comune alla singola Regione, ossia il disaster recovery multi-AZ, è presentato nel post del blog, Building highly resilient applications using Amazon Application Recovery Controller, Part 1: Single-Region stack. In questo caso la strategia adottata è quella di garantire il più possibile l'isolamento tra le zone di disponibilità, ossia come le regioni operano. Usando questa strategia alternativa puoi scegliere un approccio attivo/attivo o attivo/passivo.

> **Note**
>
> Alcuni carichi di lavoro hanno requisiti normativi di residenza dei dati. Se questo si applica a un carico di lavoro in una località che attualmente ha solo una Regione AWS, la multi-regione non soddisferà i requisiti aziendali. Le strategie con più zone di disponibilità offrono una buona protezione dalla maggior parte dei disastri.

3. Valuta le risorse del tuo carico di lavoro e quale sarà la loro configurazione nella regione di ripristino prima del failover (durante la normale operatività).

   Per l'infrastruttura e le risorse AWS, usa il modello Infrastructure as code, come AWS CloudFormation o strumenti di terze parti, come Hashicorp Terraform. Per l'implementazione in più account e regioni con una singola operazione, puoi usare AWS CloudFormation StackSets. Per le strategie multi-sito attivo/attivo e standby a caldo, l'infrastruttura distribuita nella tua regione di ripristino ha le stesse risorse della regione principale. Per le strategie Pilot Light e Warm Standby l'infrastruttura distribuita richiederà azioni aggiuntive per essere pronta per la produzione. I parametri e la logica condizionale di CloudFormation permettono di controllare se uno stack implementato è attivo o in standby con un unico modello. Quando usi Elastic Disaster Recovery, il servizio replica e orchestra il ripristino delle configurazioni delle applicazioni e delle risorse di calcolo.

   Tutte le strategie di disaster recovery richiedono l'esecuzione del backup delle origini dati all'interno della Regione AWS e la copia di tali backup nella regione di ripristino. AWS Backup offre una visualizzazione a livello centrale che consente di configurare, pianificare e monitorare i backup per tali risorse. Per Pilot Light, Warm Standby e Multi-sito attivo/attivo, devi anche replicare i dati dalla regione principale alle risorse di dati nella regione di ripristino, come le istanze DB di Amazon Relational Database Service (Amazon RDS) o le tabelle di Amazon DynamoDB. Queste risorse di dati sono pertanto attive e pronte per servire le richieste nella regione di ripristino.

   Per ulteriori informazioni sul funzionamento dei servizi AWS fra le regioni, consulta questa serie di blog sulla creazione di un'applicazione in più regioni con servizi AWS.

4. Stabilisci e implementa le modalità con cui preparerai la tua regione al failover nel momento in cui sarà necessario (durante un'emergenza).

   Per la strategia attivo/attivo multisito, il failover significa evacuare una regione e usare le regioni attive rimanenti. In generale, tali regioni sono pronte per accettare il traffico. Per le strategie Pilot Light e di Warm Standby, le azioni di ripristino devono implementare le risorse mancanti, come le istanze EC2 nella Figura 20, insieme a risorse mancanti di altro tipo.

   Per tutte le strategie precedenti potresti dover promuovere istanze di database di sola lettura a istanze di lettura/scrittura principali.

   Per il backup e il ripristino, il ripristino dei dati dai backup crea risorse per tali dati, come volumi EBS, istanze DB RDS e tabelle DynamoDB. Devi anche ripristinare l'infrastruttura e implementare il codice. Puoi usare AWS Backup per ripristinare i dati nella regione di ripristino. Per ulteriori

dettagli, consulta REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o una riproduzione dei dati dalle origini. La ricostruzione dell'infrastruttura comprende la creazione di risorse come istanze EC2, oltre ad Amazon Virtual Private Cloud (Amazon VPC), sottoreti e gruppi di sicurezza necessari. Puoi automatizzare gran parte del processo di ripristino. Per scoprire come farlo, consulta questo post del blog.

5. Stabilisci e implementa le modalità con cui reindirizzerai il traffico al failover nel momento in cui sarà necessario (durante un'emergenza).

   Questa operazione di failover può essere avviata automaticamente o manualmente. Il failover avviato automaticamente in base a controlli dell'integrità o allarmi deve essere usato con attenzione, poiché un failover non necessario (falso allarme) comporta dei costi in termini di non disponibilità e perdita dei dati. Pertanto si usa spesso il failover avviato manualmente. In questo caso, devi comunque automatizzare i passaggi del failover, in modo che l'avvio manuale si limiti al clic su un pulsante.

   Esistono diverse opzioni di gestione del traffico da considerare quando si usano i servizi AWS. Tra le opzioni, vi è l'utilizzo di Amazon Route 53. Con Amazon Route 53 puoi associare più endpoint IP in una o più Regioni AWS con un nome di dominio Route 53. Per implementare il failover avviato manualmente, puoi utilizzare Amazon Application Recovery Controller, che fornisce un'API del piano dati a elevata disponibilità per reinstradare il traffico verso la Regione di ripristino. Nella fase di implementazione del failover, usa le operazioni di piano dati ed evita quelle del piano di controllo (control-plane), come illustrato in REL11-BP04 Fare affidamento al piano dati invece che al piano di controllo (control-plane) durante il ripristino.

   Per ulteriori informazioni su questa e altre opzioni, consulta questa sezione del whitepaper sul ripristino di emergenza.

6. Progetta un piano per il failback del carico di lavoro.

   Si parla di failback quando un'operazione del carico di lavoro torna alla regione principale, dopo che un'emergenza è diminuita di intensità. Il provisioning di infrastruttura e codice alla regione principale in genere segue gli stessi passaggi usati inizialmente, affidandosi al modello Infrastructure as code e alle pipeline di implementazione del codice. La sfida del failback è il ripristino dei data store e la garanzia della loro coerenza con la regione di ripristino attiva.

   Nello stato di failover i database nella regione di ripristino sono attivi e hanno dati aggiornati. L'obiettivo è eseguire una nuova sincronizzazione tra la regione di ripristino e la regione principale, per garantire il suo aggiornamento.

Alcuni servizi AWS eseguono questa operazione in automatico. In caso di utilizzo delle tabelle globali Amazon DynamoDB, anche se la tabella nella regione principale era diventata non disponibile, quando torna di nuovo online, ripristina la propagazione di scritture in sospeso. Se utilizzi il Database globale Amazon Aurora e un failover pianificato gestito, viene mantenuta la topologia di replica esistente del database globale Aurora. Pertanto, l'istanza precedente in lettura/scrittura nella regione principale diventa una replica e riceve gli aggiornamenti dalla regione di ripristino.

Nei casi in cui questo non è automatico devi ristabilire il database nella regione principale come replica del database nella regione di ripristino. In molti casi questo comporterà l'eliminazione del database principale precedente e la creazione di nuove repliche.

Dopo un failover, se puoi proseguire l'esecuzione nella tua regione di ripristino, valuta la possibilità di farlo nella tua regione principale. Effettueresti comunque tutte le operazioni precedenti per trasformare la precedente regione principale in una regione di ripristino. Alcune organizzazioni eseguono una rotazione pianificata, scambiando periodicamente le regioni principale e di ripristino (ad esempio, ogni tre mesi).

Tutti i passaggi richiesti per failover e failback devono essere inseriti in un playbook disponibile a tutti i membri del team, sottoposto periodicamente a revisione.

Se usi Elastic Disaster Recovery, il servizio fornirà assistenza per l'orchestrazione e l'automazione del processo di failback. Per ulteriori informazioni, consulta Performing a failback.

Livello di impegno per il piano di implementazione: elevato

Risorse

Best practice correlate:

- the section called “REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o una riproduzione dei dati dalle origini”
- the section called “REL11-BP04 Fare affidamento al piano dati invece che al piano di controllo (control-plane) durante il ripristino”
- the section called “REL13-BP01 Definizione degli obiettivi di ripristino in caso di downtime e perdita di dati”

Documenti correlati:

- AWS Architecture Blog: serie sul disaster recovery
- Ripristino di emergenza dei carichi di lavoro su AWS: ripristino nel cloud (whitepaper di AWS)
- Opzioni di disaster recovery nel cloud
- Build a serverless multi-region, active-active backend solution in an hour
- Multi-region serverless backend — reloaded
- RDS: creazione di una replica di lettura in un fra le regioni
- Route 53: configurazione del failover DNS
- S3: replica tra regioni
- Cosa è AWS Backup?
- What is Amazon Application Recovery Controller?
- AWS Elastic Disaster Recovery
- HashiCorp Terraform: Get Started - AWS
- Partner APN: partner che possono assistere con il disaster recovery
- Marketplace AWS: prodotti utilizzabili per il disaster recovery

Video correlati:

- Disaster Recovery of Workloads on AWS
- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (ARC209-R2)
- Get Started with AWS Elastic Disaster Recovery | Amazon Web Services

REL13-BP03 Esecuzione di test sull'implementazione del disaster recovery per convalidare l'implementazione

Testa regolarmente il failover nel sito di ripristino per verificare che funzioni correttamente e che sia possibile soddisfare l'RTO e l'RPO.

Anti-pattern comuni:

- Non eseguire mai failover di prova in produzione.

Vantaggi dell'adozione di questa best practice: testare regolarmente il piano di disaster recovery verifica che funzioni quando necessario e che il tuo team sappia come eseguire la strategia.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Un modello da evitare è lo sviluppo di percorsi di ripristino eseguiti raramente. Ad esempio, è possibile che si disponga di un archivio dati secondario utilizzato per query di sola lettura. Quando scrivi in un archivio dati e quello principale ha un guasto, puoi eseguire il failover verso l'archivio dati secondario. Se non testi frequentemente questo failover, è possibile che i presupposti relativi alle funzionalità dell'archivio dati secondario non siano corretti. La capacità dell'archivio dati secondario, che potrebbe essere stata sufficiente durante l'ultimo test, potrebbe non essere più in grado di tollerare il carico in questo scenario. La nostra esperienza ha dimostrato che l'unico ripristino da errore che funziona è il percorso sottoposto a frequenti test. Per questo è preferibile avere un numero ridotto di percorsi di ripristino. Puoi stabilire dei modelli di ripristino e testarli regolarmente. Se disponi di un percorso di ripristino complesso o critico, devi comunque riprodurre regolarmente il guasto specifico in produzione per convincerti che il percorso di ripristino funzioni. Nell'esempio appena discusso, è necessario eseguire il failover regolarmente in standby, indipendentemente dalle necessità.

Passaggi dell'implementazione

1. Progetta i carichi di lavoro per il ripristino. Esegui regolarmente test dei tuoi percorsi di ripristino. Il calcolo orientato al ripristino identifica le caratteristiche nei sistemi che migliorano il ripristino: isolamento e ridondanza, ripristino a livello di sistema dello stato precedente rispetto alle modifiche, capacità di fornire diagnostica, ripristino automatico, progettazione modulare e possibilità di riavvio. Prova il percorso di ripristino per verificare di poter completare il ripristino nel tempo specificato e in base allo stato specificato. Usa i tuoi runbook durante questo ripristino per documentare i problemi e trovarne le soluzioni prima del test successivo.
2. Per i carichi di lavoro basati su Amazon EC2, utilizza AWS Elastic Disaster Recovery per implementare e avviare istanze di esercitazione per la tua strategia di disaster recovery. AWS Elastic Disaster Recovery consente di eseguire esercitazioni in modo efficiente, per prepararsi per un evento di failover. Puoi anche avviare spesso le istanze usando Elastic Disaster Recovery per scopi di test ed esercitazione senza reindirizzare il traffico.

Risorse

Documenti correlati:

- Partner APN: partner che possono assistere con il ripristino di emergenza
- AWS Architecture Blog: serie sul disaster recovery
- Marketplace AWS: prodotti utilizzabili per il disaster recovery
- AWS Elastic Disaster Recovery
- Ripristino di emergenza dei carichi di lavoro su AWS: ripristino nel cloud (whitepaper di AWS)
- AWS Elastic Disaster Recovery Preparing for Failover
- The Berkeley/Stanford recovery-oriented computing project
- What is AWS Fault Injection Simulator?

Video correlati:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications
- AWS re:Invent 2019: Backup-and-restore and disaster-recovery solutions with AWS

REL13-BP04 Gestione della deviazione di configurazione nel sito o nella regione del disaster recovery

Per eseguire una procedura di disaster recovery (DR), il carico di lavoro deve essere in grado di riprendere le normali operazioni in modo tempestivo, senza perdite rilevanti di funzionalità o di dati, una volta che l'ambiente di DR è stato messo online. Per raggiungere questo obiettivo, è essenziale mantenere coerenti l'infrastruttura, i dati e le configurazioni tra l'ambiente di disaster recovery e l'ambiente primario.

Risultato desiderato: la configurazione e i dati del sito di disaster recovery sono identici a quelli del sito primario, il che facilita un ripristino rapido e completo in caso di necessità.

Anti-pattern comuni:

- Non riesci ad aggiornare le posizioni di ripristino quando vengono apportate modifiche alle posizioni primarie. Questo determina configurazioni obsolete che potrebbero ostacolare gli sforzi di ripristino.
- Non consideri le potenziali limitazioni, come le differenze di servizio tra le posizioni primaria e di ripristino, che possono portare a errori imprevisti durante il failover.
- Per l'aggiornamento e la sincronizzazione dell'ambiente di disaster recovery fai riferimento a processi manuali, il che aumenta il rischio di errore umano e di incoerenza.

- Non riesci a rilevare la deviazione di configurazione. Questo ti porta falsamente a pensare che il sito di disaster recovery sia pronto prima di un incidente.

Vantaggi dell'adozione di questa best practice: la coerenza tra l'ambiente di disaster recovery e l'ambiente primario migliora notevolmente le probabilità della riuscita di un ripristino dopo un incidente e riduce il rischio di errore della procedura di ripristino.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Un approccio completo alla gestione della configurazione e alla preparazione al failover può aiutarti a verificare che il sito di disaster recovery sia costantemente aggiornato e pronto a subentrare in caso di errore del sito primario.

Per ottenere la coerenza tra l'ambiente primario e quello di disaster recovery (DR), verifica che le pipeline di distribuzione distribuiscano le applicazioni sia al sito primario che a quello di disaster recovery. Implementa le modifiche ai siti di disaster recovery dopo un adeguato periodo di valutazione (noto anche come implementazioni distribuite) per rilevare i problemi nel sito primario e arrestare l'implementazione prima che si diffondano. Implementa il monitoraggio per rilevare la deviazione della configurazione e tenere traccia delle modifiche e della conformità negli ambienti. Esegui la correzione automatica nel sito di disaster recovery per mantenerlo completamente coerente e pronto a subentrare in caso di incidente.

Passaggi dell'implementazione

1. Verifica che la Regione di disaster recovery contenga i servizi AWS e le funzionalità richieste per una corretta esecuzione del piano di disaster recovery.
2. Utilizza infrastructure as code (IaC). Mantieni accurati i modelli di configurazione dell'infrastruttura di produzione e dell'applicazione e applicali regolarmente all'ambiente di disaster recovery. AWS CloudFormation è in grado di rilevare le deviazioni tra ciò che i modelli CloudFormation specificano e ciò che viene effettivamente distribuito.
3. Configura le pipeline CI/CD per distribuire le applicazioni e gli aggiornamenti dell'infrastruttura in tutti gli ambienti, compresi i siti primari e di disaster recovery. Soluzioni CI/CD come AWS CodePipeline possono automatizzare il processo di implementazione, riducendo il rischio di deviazione della configurazione.
4. Implementazioni distribuite tra gli ambienti primario e di disaster recovery. Questo approccio consente di distribuire e testare inizialmente gli aggiornamenti nell'ambiente primario, isolando

così i problemi nel sito primario prima che vengano propagati al sito di disaster recovery. Questo approccio impedisce che i difetti vengano inviati contemporaneamente alla produzione e al sito di disaster recovery e mantiene l'integrità dell'ambiente di disaster recovery.

5. Monitora costantemente le configurazioni delle risorse sia nell'ambiente primario che in quello di disaster recovery. Soluzioni come AWS Config possono aiutare a far rispettare la conformità della configurazione e a rilevare eventuali deviazioni, contribuendo a mantenere configurazioni coerenti tra gli ambienti.
6. Implementa meccanismi di avviso per monitorare e notificare qualsiasi deviazione della configurazione o interruzione o ritardo nella replica dei dati.
7. Automatizza la correzione delle deviazioni di configurazione rilevate.
8. Pianifica audit periodici e controlli di conformità per verificare l'allineamento continuo tra le configurazioni primaria e di disaster recovery. Le revisioni periodiche aiutano a mantenere la conformità con le regole definite e a identificare eventuali discrepanze che devono essere risolte.
9. Verifica eventuali discordanze nella capacità allocata da AWS, in Service Quotas, nelle limitazioni (della larghezza di banda della rete) e nelle discrepanze di configurazione e versione.

Risorse

Best practice correlate:

- REL01-BP01 Consapevolezza su Service Quotas e vincoli di servizio
- REL01-BP02 Gestione delle Service Quotas in più account e Regioni
- REL01-BP04 Monitoraggio e gestione delle quote
- REL13-BP03 Esecuzione di test sull'implementazione del disaster recovery per convalidare l'implementazione

Documenti correlati:

- Remediating Noncompliant AWS Resources by Regole di AWS Config
- AWS Systems Manager Automation
- AWS CloudFormation: Detecting unmanaged configuration changes to stacks and resources
- AWS CloudFormation: Detect Drift on an Entire CloudFormation Stack
- AWS Systems Manager Automation
- Ripristino di emergenza dei carichi di lavoro su AWS: ripristino nel cloud (whitepaper di AWS)

- In che modo è possibile implementare una soluzione di gestione della configurazione dell'infrastruttura in AWS?
- Remediating Noncompliant AWS Resources by Regole di AWS Config

Video correlati:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (ARC209-R2)

Esempi correlati:

- CloudFormation Registry
- Quota Monitor for AWS
- Implement automatic drift remediation for AWS CloudFormation using Amazon CloudWatch and AWS Lambda
- AWS Architecture Blog: serie sul disaster recovery
- Marketplace AWS: prodotti utilizzabili per il disaster recovery
- Automatizzazione di distribuzioni pratiche e sicure

REL13-BP05 Automatizzazione del ripristino

Implementa meccanismi di ripristino testati e automatizzati che siano affidabili, osservabili e riproducibili per ridurre il rischio e l'impatto aziendale di guasti ed errori.

Risultato desiderato: hai implementato un flusso di lavoro di automazione ben documentato, standardizzato e accuratamente testato per i processi di ripristino. L'automazione del ripristino corregge automaticamente i problemi secondari che comportano un basso rischio di perdita di dati o di indisponibilità. Puoi invocare rapidamente i processi di ripristino per gli incidenti gravi, osservare il comportamento della correzione durante il loro funzionamento e terminare i processi se osservi situazioni pericolose o errori.

Anti-pattern comuni:

- Dipendi da componenti o meccanismi che si trovano in uno stato non riuscito o danneggiato come parte del piano di ripristino.

- I processi di ripristino richiedono un intervento manuale, come l'accesso alla console (noto anche come ClickOps).
- Avvii le procedure di ripristino automaticamente in situazioni che presentano un rischio elevato di perdita o indisponibilità dei dati.
- Non includi un meccanismo per interrompere una procedura di ripristino (come un cavo Andon o un grande pulsante rosso di arresto) che non funziona o che comporta rischi aggiuntivi.

Vantaggi dell'adozione di questa best practice:

- Maggiore affidabilità, prevedibilità e coerenza delle operazioni di ripristino.
- Capacità di soddisfare obiettivi di ripristino più rigorosi, tra cui Obiettivo del tempo di ripristino (RTO) e Obiettivo del punto di ripristino (RPO).
- Riduzione della probabilità di non riuscita del ripristino durante un incidente.
- Riduzione del rischio di errori associati a processi di ripristino manuali, soggetti a errori umani.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Per implementare il ripristino automatizzato, è necessario un approccio completo che utilizzi i servizi AWS e le best practice. Per iniziare, identifica i componenti critici e i potenziali punti di errore nel carico di lavoro. Sviluppa processi automatizzati in grado di ripristinare i carichi di lavoro e i dati in caso di errori senza l'intervento umano.

Sviluppa l'automazione del ripristino utilizzando i principi infrastructure as code (IaC). In questo modo l'ambiente di ripristino è coerente con l'ambiente di origine e consente il controllo delle versioni dei processi di ripristino. Per orchestrare flussi di lavoro di ripristino complessi, valuta soluzioni come AWS Systems Manager Automations o AWS Step Functions.

L'automazione dei processi di ripristino offre vantaggi significativi e può aiutare a raggiungere più facilmente Obiettivo del tempo di ripristino (RTO) e Obiettivo del punto di ripristino (RPO). Tuttavia, si possono verificare situazioni impreviste che possono causare un esito negativo o creare nuovi rischi, come tempo di inattività aggiuntivo e perdita di dati. Per ridurre questo rischio, occorre offrire la possibilità di interrompere rapidamente un'automazione dei ripristino in corso. Una volta interrotta, si può indagare e adottare misure correttive.

Per i carichi di lavoro supportati, valuta soluzioni come AWS Elastic Disaster Recovery (AWS DRS) per fornire un failover automatico. AWS DRS replica continuamente le macchine (compresi sistema operativo, configurazione dello stato del sistema, database, applicazioni e file) in un'area di gestione temporanea nell'Account AWS di destinazione e nella Regione preferita. Se si verifica un incidente, AWS DRS automatizza la conversione dei server replicati in carichi di lavoro completamente allocati nella Regione di ripristino su AWS.

La manutenzione e il miglioramento del ripristino automatico sono un processo continuo. Verifica e perfeziona continuamente le procedure di ripristino in base alle lezioni apprese e rimani aggiornato sui nuovi servizi e funzionalità AWS che possono migliorare le capacità di ripristino.

Passaggi dell'implementazione

1. Pianifica il ripristino automatico
    a. Esegui una revisione approfondita dell'architettura, dei componenti e delle dipendenze del carico di lavoro per identificare e pianificare i meccanismi di ripristino automatico. Classifica le dipendenze del carico di lavoro in dipendenze hard e soft. Le dipendenze hard sono quelle senza le quali il carico di lavoro non può funzionare e per le quali non è possibile fornire un sostituto. Le dipendenze soft sono quelle utilizzate abitualmente dal carico di lavoro, ma che possono essere sostituite da sistemi o processi sostitutivi temporanei o che possono essere gestite con una degradazione regolare.
    b. Stabilisci processi per identificare e recuperare i dati mancanti o danneggiati.
    c. Definisci i passaggi per confermare lo stato stazionario ripristinato dopo il completamento delle azioni di ripristino.
    d. Prendi in considerazione tutte le azioni necessarie per rendere il sistema ripristinato pronto per il servizio completo, come il pre-riscaldamento e la compilazione delle cache.
    e. Considera i problemi che si potrebbero verificare durante il processo di ripristino e come individuarli e correggerli.
    f. Considera gli scenari in cui il sito primario e il relativo piano di controllo (control-plane) non sono accessibili. Verifica che le azioni di ripristino possano essere eseguite in modo indipendente senza ricorso al sito primario. Considera soluzioni come Amazon Application Recovery Controller (ARC) per reindirizzare il traffico senza dover modificare manualmente i record DNS.
2. Sviluppa un processo di ripristino automatico
    a. Implementa il rilevamento automatico dei guasti e meccanismi di failover per un ripristino automatico. Crea dashboard, ad esempio con Amazon CloudWatch, per segnalare lo stato di avanzamento e lo stato di integrità delle procedure di ripristino automatiche. Includi procedure

per convalidare le operazioni di ripristino riuscite. Fornisci un meccanismo per interrompere un ripristino in corso.

b. Crea playbook come processo di fallback per guasti che non possono essere ripristinati automaticamente e prendi in considerazione il piano di disaster recovery.

c. Esegui il test dei processi di ripristino come descritto in REL13-BP03.

3. Preparati per il ripristino

a. Valuta lo stato del sito di ripristino e distribuisci in anticipo i componenti critici. Per ulteriori dettagli, consulta REL13-BP04.

b. Definisci ruoli, responsabilità e processi decisionali chiari per le operazioni di ripristino, coinvolgendo le parti interessate e i team dell'organizzazione.

c. Definisci le condizioni per avviare i processi di ripristino.

d. Crea un piano per invertire il processo di ripristino e tornare al sito primario, se richiesto o dopo che è stato considerato sicuro.

Risorse

Best practice correlate:

- REL07-BP01 Utilizzo dell'automazione per l'acquisizione o il dimensionamento delle risorse
- REL11-BP01 Monitoraggio di tutti i componenti del carico di lavoro per la rilevazione dei guasti
- REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino
- REL13-BP03 Esecuzione di test sull'implementazione del disaster recovery per convalidare l'implementazione
- REL13-BP04 Gestione della deviazione di configurazione nel sito o nella Regione del disaster recovery

Documenti correlati:

- AWS Architecture Blog: serie sul disaster recovery
- Ripristino di emergenza dei carichi di lavoro su AWS: ripristino nel cloud (whitepaper di AWS)
- Orchestrate Disaster Recovery Automation using Amazon Route 53 ARC and AWS Step Functions
- Build AWS Systems Manager Automation runbooks using AWS CDK
- Marketplace AWS: prodotti utilizzabili per il disaster recovery
- AWS Systems Manager Automation

- AWS Elastic Disaster Recovery
- Using Elastic Disaster Recovery for Failover and Failback
- Risorse di AWS Elastic Disaster Recovery
- Partner APN: partner che possono assistere con il disaster recovery

Video correlati:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (ARC209-R2)
- AWS re:Invent 2022: AWS On Air ft. AWS Failback for AWS Elastic Disaster Recovery

Efficienza delle prestazioni

Il pilastro dell'efficienza delle prestazioni include la capacità di utilizzare in modo efficiente le risorse nel cloud per soddisfare i requisiti in termini di prestazione e di mantenere tale efficienza a fronte al cambiamento della domanda e all'evoluzione delle tecnologie. Puoi trovare linee guida prescrittive sull'implementazione nel whitepaper sul pilastro dell'efficienza delle prestazioni.

Aree delle best practice

- Scelta dell'architettura
- Calcolo e hardware
- Gestione dei dati
- Reti e distribuzione di contenuti
- Processo e cultura

Scelta dell'architettura

Questions

- PERF 1. In che modo selezioni le risorse e l'architettura cloud appropriate per il tuo carico di lavoro?

PERF 1. In che modo selezioni le risorse e l'architettura cloud appropriate per il tuo carico di lavoro?

La soluzione ottimale per un determinato carico di lavoro può variare e le soluzioni spesso combinano molteplici approcci. I carichi di lavoro Well-Architected utilizzano soluzioni multiple e forniscono funzionalità diverse per migliorare le prestazioni.

Best practice

- PERF01-BP01 Informazioni e identificazione dei servizi e delle funzionalità cloud disponibili
- PERF01-BP02 Utilizzo delle indicazioni del provider cloud o di un partner appropriato per conoscere gli schemi di architettura e le best practice
- PERF01-BP03 Fattore di costo nelle decisioni architettoniche
- PERF01-BP04 Valutazione dell'influenza dei compromessi sui clienti e sull'efficienza dell'architettura
- PERF01-BP05 Usa politiche e architetture di riferimento
- PERF01-BP06 Uso del benchmarking per guidare le decisioni sull'architettura
- PERF01-BP07 Uso di un approccio basato sui dati per le scelte dell'architettura

PERF01-BP01 Informazioni e identificazione dei servizi e delle funzionalità cloud disponibili

Informati continuamente e identifica i servizi e le configurazioni disponibili che ti aiutano a prendere le decisioni giuste sull'architettura e a migliorare l'efficienza delle prestazioni dei carichi di lavoro.

Anti-pattern comuni:

- Utilizzi il cloud come data center in co-location.
- Non stai modernizzando la tua applicazione con la migrazione al cloud.
- Stai solo usando un tipo di archiviazione per tutte le cose che devono essere conservate in modo persistente.
- Se necessario, utilizzi tipi di istanze strettamente correlate ai tuoi standard attuali, ma più grandi.
- Distribuisci e gestisci le tecnologie disponibili come servizi gestiti.

Vantaggi dell'adozione di questa best practice: prendendo in considerazione nuovi servizi e configurazioni, puoi migliorare notevolmente le prestazioni, ridurre i costi e ottimizzare le attività

necessarie per mantenere il carico di lavoro. Puoi anche accelerare il time-to-value per i prodotti abilitati al cloud.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

AWS rilascia continuamente nuovi servizi e funzionalità in grado di migliorare le prestazioni e ridurre i costi dei carichi di lavoro del cloud. Rimanere aggiornati su questi nuovi servizi e funzionalità è fondamentale per mantenere l'efficacia delle prestazioni nel cloud. La modernizzazione dell'architettura dei carichi di lavoro consente inoltre di accelerare la produttività, promuovere l'innovazione e sbloccare ulteriori opportunità di crescita.

Passaggi dell'implementazione

- Esegui l'inventario del software e dell'architettura del carico di lavoro per i servizi correlati. Determina su quale categoria di prodotti ottenere ulteriori informazioni.
- Esplora le offerte AWS per individuare e conoscere i servizi e le opzioni di configurazione pertinenti che possono aiutarti a migliorare le prestazioni e ridurre i costi e la complessità operativa.
  - Amazon Web Services Cloud
  - AWS Academy
  - Novità di AWS
  - Blog AWS
  - AWS Skill Builder
  - Eventi e webinar AWS
  - AWS Training e certificazioni
  - Canale YouTube di AWS
  - Workshop AWS
  - Community AWS
- Usa Amazon Q per ricevere informazioni e consigli pertinenti sui servizi.
- Usa gli ambienti sandbox non di produzione per comprendere e sperimentare nuovi servizi senza incorrere in costi aggiuntivi.
- Scopri servizi e funzionalità cloud sempre nuovi.

Risorse

Documenti correlati:

- Overview of Amazon Web Services
- Caratteristiche di Amazon EC2
- Impara passo per passo con il Programma di apprendimento dei Partner AWS
- Formazione e certificazione AWS
- My learning path to become an AWS solutions architect
- AWS Architecture Center
- AWS Partner Network
- AWS Biblioteca di soluzioni di
- Centro conoscenze di AWS
- Costruisci applicazioni moderne su AWS

Video correlati:

- AWS re:Invent 2023 - What's new with Amazon EC2
- AWS re:Invent 2022 - Reduce your operational and infrastructure costs with Amazon ECS
- AWS re:Invent 2023 - Build with the efficiency, agility & innovation of the cloud with AWS
- AWS re:Invent 2022 - Deploy ML models for inference at high performance and low cost
- This is my Architecture

Esempi correlati:

- AWS Esempi di
- AWS Esempi di SDK

PERF01-BP02 Utilizzo delle indicazioni del provider cloud o di un partner appropriato per conoscere gli schemi di architettura e le best practice

Usa le risorse aziendali del cloud come documentazione, solutions architect, servizi professionali o partner appropriati per guidare le tue decisioni sull'architettura. Queste risorse ti aiutano a rivedere e migliorare l'architettura per ottenere prestazioni ottimali.

Anti-pattern comuni:

- AWS è usato come un comune provider di servizi cloud.
- I servizi AWS vengono utilizzati in modo diverso rispetto alla loro progettazione iniziale.
- Le indicazioni vengono seguite senza considerare il contesto aziendale.

Vantaggi dell'adozione di questa best practice: avvalersi della guida di un provider di servizi cloud o di un partner appropriato può aiutarti a fare le scelte giuste per l'architettura del tuo carico di lavoro e darti fiducia nelle tue decisioni.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

AWS offre un'ampia scelta di linee guida, documentazione e risorse che possono aiutarti a creare e gestire i carichi di lavoro del cloud in modo efficiente. La documentazione AWS fornisce esempi di codice, esercitazioni e spiegazioni dettagliate sui servizi. Oltre alla documentazione, AWS offre programmi di formazione e certificazione, solutions architect e servizi professionali che i clienti possono usare per esplorare diversi aspetti dei servizi cloud e implementare un'architettura cloud efficiente su AWS.

Sfrutta queste risorse per ottenere approfondimenti sulle informazioni e sulle best practice preziose per risparmiare tempo e ottenere risultati migliori nel Cloud AWS.

Passaggi dell'implementazione

- Consulta la documentazione e le linee guida AWS e segui le best practice. Queste risorse possono aiutarti a scegliere e configurare i servizi in modo efficace e a ottenere prestazioni migliori.
  - Documentazione di AWS (come guide utente e whitepaper)
  - Blog AWS
  - AWS Training e certificazioni
  - Canale YouTube di AWS
- Partecipa agli eventi per i partner AWS (come summit AWS a livello mondiale, gruppi di utenti di AWS re:Invent e workshop) per apprendere dagli esperti AWS le best practice per l'utilizzo dei servizi AWS.
  - Impara passo per passo con il Programma di apprendimento dei Partner AWS
  - Eventi e webinar AWS

  - Workshop AWS
  - Community AWS
- Contatta AWS per ricevere assistenza quando ti occorrono ulteriori indicazioni o informazioni sui prodotti. AWS I Solutions Architect e i servizi professionali di AWS forniscono indicazioni per l'implementazione delle soluzioni. AWS I partner mettono a disposizione la propria conoscenza di AWS per aiutarti ad assicurare alla tua azienda agilità e innovazione.
- Usa Supporto se hai bisogno di supporto tecnico per utilizzare un servizio in modo efficace. I nostri piani di supporto sono pensati per offrirti il giusto mix di strumenti e competenze in modo da poter conseguire il successo con AWS ottimizzando le prestazioni, gestendo i rischi e tenendo sotto controllo i costi.

Risorse

Documenti correlati:

- AWS Architecture Center
- AWS Partner Network
- Biblioteca di soluzioni di AWS
- Centro conoscenze di AWS
- Supporto AWS Enterprise

Video correlati:

- This is my Architecture
- AWS re:Invent 2023 - Advanced event-driven patterns with Amazon EventBridge
- AWS re:Invent 2023 - Implementing distributed design patterns on AWS
- AWS re:Invent 2023 - Application architecture as code

Esempi correlati:

- AWS Esempi di
- Esempi di SDK AWS
- AWS Analytics Reference Architecture

PERF01-BP03 Fattore di costo nelle decisioni architettoniche

Tieni conto dei costi nelle decisioni sull'architettura per migliorare l'utilizzo delle risorse e l'efficienza delle prestazioni del tuo carico di lavoro cloud. Quando si è consapevoli delle implicazioni dei costi del carico di lavoro cloud, è più probabile che si utilizzino risorse efficienti e si riducano le procedure inutili.

Anti-pattern comuni:

- Utilizzi una sola famiglia di istanze.
- Ometti di valutare le soluzioni con licenza rispetto alle soluzioni open-source.
- Non definisci le policy del ciclo di vita dell'archiviazione.
- Non recensisci i nuovi servizi e funzionalità di. Cloud AWS
- Utilizzi solo lo storage a blocchi.

Vantaggi dell'adozione di questa best practice: la contabilizzazione dei costi nel processo decisionale consente di utilizzare risorse più efficienti ed esplorare altri investimenti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

L'ottimizzazione dei carichi di lavoro in base ai costi può migliorare l'utilizzo delle risorse ed evitare sprechi nel carico di lavoro cloud. Tenere conto dei costi nelle decisioni sull'architettura di solito include il corretto dimensionamento dei componenti del carico di lavoro e l'abilitazione dell'elasticità, comportando una migliore efficienza delle prestazioni del carico di lavoro cloud.

Passaggi dell'implementazione

- Stabilisci gli obiettivi di costo, come i limiti del budget, per il tuo carico di lavoro cloud.
- Identifica i componenti chiave, come istanze e archiviazione, che determinano il costo del carico di lavoro. Puoi usare Calcolatore dei prezzi AWS e AWS Cost Explorer per identificare i principali fattori di costo del carico di lavoro.
- Esamina i modelli di prezzo nel cloud, ad esempio istanze on-demand, riservate, Savings Plans e istanze spot.
- Segui le best practice per l'ottimizzazione dei costi di Well-Architected per ottimizzare questi componenti principali in termini di costi.

- Monitora e analizza continuamente i costi per identificare le opportunità di ottimizzazione dei costi nel tuo carico di lavoro.
  - Usa Budget AWS per ricevere gli avvisi per i costi inaccettabili.
  - Usa AWS Compute Optimizer o AWS Trusted Advisor per ottenere suggerimenti sull'ottimizzazione dei costi.
  - Usa AWS Cost Anomaly Detection per rilevare in modo automatico le anomalie dei costi e analizzare la causa principale.

Risorse

Documenti correlati:

- Che cos'è AWS Billing and Cost Management?
- Ottimizzazione dei costi con AWS
- Scelta di una strategia di gestione dei AWS costi
- Una guida per principianti alla gestione AWS dei costi
- A Detailed Overview of the Cost Intelligence Dashboard
- AWS Architecture Center
- Biblioteca di soluzioni di AWS
- Centro conoscenze di AWS

Video correlati:

- This is my Architecture
- AWS re:Invent 2023 - Cosa c'è di nuovo con l'ottimizzazione dei costi AWS
- AWS re:Invent 2023 - Ottimizza costi e prestazioni e monitora i progressi verso la mitigazione
- AWS re:Invent 2023 - best practice per l'ottimizzazione dei costi di storage AWS
- AWS re:Invent 2023 - Ottimizza i costi nei tuoi ambienti con più account

Esempi correlati:

- AWS Compute Optimizer Codice demo
- Cost Optimization Workshop

- Cloud Financial Management Technical Implementation Playbooks
- Startup optimization: Tuning application performance for maximum efficiency
- Serverless Optimization Workshop (Performance and Cost)
- Scaling cost effective architectures

PERF01-BP04 Valutazione dell'influenza dei compromessi sui clienti e sull'efficienza dell'architettura

Quando valuti i miglioramenti correlati alle prestazioni, determina quali scelte hanno impatto sui clienti e sull'efficienza del carico di lavoro. Ad esempio, se l'utilizzo di un datastore chiave-valore aumenta le prestazioni del sistema, è importante valutare in che modo la consistenza finale intrinseca di questo cambiamento avrà un impatto sui clienti.

Anti-pattern comuni:

- Ritieni che tutti i vantaggi prestazionali debbano essere implementati, anche se ci sono compromessi per l'implementazione.
- Valuti di apportare modifiche ai carichi di lavoro solo quando un problema prestazionale ha raggiunto un punto critico.

Vantaggi dell'adozione di questa best practice: quando si valutano potenziali miglioramenti relativi alle prestazioni, è necessario decidere se i compromessi per le modifiche sono accettabili con i requisiti del carico di lavoro. In alcuni casi, potrebbe essere necessario implementare controlli aggiuntivi per compensare i compromessi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Identifica le aree critiche della tua architettura in termini di prestazioni e impatto sui clienti. Stabilisci in che modo puoi apportare miglioramenti e quali compromessi comportano, oltre al loro impatto sul sistema e sull'esperienza degli utenti. L'implementazione di cache di dati, ad esempio, può contribuire a migliorare notevolmente le prestazioni ma richiede una strategia ben definita sulle modalità e sui tempi di aggiornamento o di invalidamento dei dati che vi sono contenuti, per evitare che il sistema si comporti in modo non corretto.

Passaggi dell'implementazione

- Comprendi i requisiti del tuo carico di lavoro e i contratti sul livello di servizio (SLA).

- Definisci chiaramente i fattori di valutazione. I fattori possono riguardare il costo, l'affidabilità, la sicurezza e le prestazioni del carico di lavoro.
- Seleziona l'architettura e i servizi in grado di soddisfare le tue esigenze.
- Effettua sperimentazioni e proof of concept (POC) per valutare i fattori di compromesso, l'impatto sui clienti e l'efficienza dell'architettura. Di solito, i carichi di lavoro altamente disponibili, performanti e sicuri consumano più risorse cloud offrendo al contempo una esperienza cliente migliore. Comprendi i compromessi in termini di complessità, prestazioni e costi del tuo carico di lavoro. In genere, dare la priorità a due fattori va a scapito del terzo.

Risorse

Documenti correlati:

- Amazon Builders' Library
- KPI di Quick
- Amazon CloudWatch RUM
- Documentazione di X-Ray
- Understand resiliency patterns and trade-offs to architect efficiently in the cloud

Video correlati:

- Optimize applications through Amazon CloudWatch RUM
- AWS re:Invent 2023 - Capacity, availability, cost efficiency: Pick three
- AWS re:Invent 2023 - Advanced integration patterns & trade-offs for loosely coupled systems

Esempi correlati:

- Misurazione dei tempi di caricamento delle pagine con Amazon CloudWatch Synthetics
- Client Web Amazon CloudWatch RUM

PERF01-BP05 Usa politiche e architetture di riferimento

Utilizza le policy interne e le architetture di riferimento esistenti per la selezione dei servizi e delle configurazioni per una maggiore efficienza nella progettazione e nell'implementazione del carico di lavoro.

Anti-pattern comuni:

- Usi una vasta gamma di tecnologie che possono influire sul sovraccarico di gestione della tua azienda.

Vantaggi dell'adozione di questa best practice: la definizione di una policy per la scelta dell'architettura, della tecnologia e del fornitore consente di prendere decisioni rapidamente.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Avere policy interne nella selezione delle risorse e dell'architettura fornisce standard e linee guida da seguire quando si effettuano scelte architettoniche. Queste linee guida semplificano il processo decisionale nella scelta del servizio cloud giusto e possono contribuire a migliorare l'efficienza delle prestazioni. Implementi il carico di lavoro utilizzando policy o architetture di riferimento. Integra i servizi nell'implementazione cloud, quindi utilizza i test delle prestazioni per verificare che i requisiti prestazionali siano sempre rispettati.

Passaggi dell'implementazione

- Comprendi chiaramente i requisiti del tuo carico di lavoro cloud.
- Rivedi le policy interne ed esterne per identificare quelle più pertinenti.
- Utilizza le architetture di riferimento appropriate fornite dalle best practice AWS o di settore.
- Crea un contesto composto da policy, standard, architetture di riferimento e linee guida prescrittive per situazioni comuni. In questo modo i tuoi team possono muoversi più velocemente. Personalizza le risorse per il tuo settore verticale, se applicabile.
- Convalida queste policy e architetture di riferimento per il tuo carico di lavoro in ambienti sandbox.
- Resta up-to-date conforme agli standard e agli AWS aggiornamenti del settore per assicurarti che le tue policy e le architetture di riferimento contribuiscano a ottimizzare il carico di lavoro sul cloud.

Risorse

Documenti correlati:

- AWS Architecture Center
- AWS Partner Network
- Biblioteca di soluzioni di AWS

- Centro conoscenze di AWS
- AWS Blog di architettura

Video correlati:

- This is my Architecture
- AWS re:Invent 2022 - Accelera il valore della tua azienda con SAP un'architettura di riferimento AWS

Esempi correlati:

- Esempi AWS
- AWS SDKEsempi

PERF01-BP06 Uso del benchmarking per guidare le decisioni sull'architettura

Esegui il benchmark delle prestazioni di un carico di lavoro esistente per comprendere le prestazioni sul cloud e guidare le decisioni sull'architettura basate sui dati.

Anti-pattern comuni:

- Fai affidamento su valori di riferimento comuni che non sono indicativi delle caratteristiche del carico di lavoro.
- L'unico punto di riferimento è dato dal feedback e dalle percezioni dei clienti.

Vantaggi dell'adozione di questa best practice: misurazione dei miglioramenti in termini di prestazioni grazie al benchmarking dell'implementazione attuale.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Utilizza test sintetici di benchmarking per valutare le prestazioni dei componenti durante il carico di lavoro. Di solito, i benchmark sono più rapidi da configurare rispetto ai test di carico e vengono utilizzati per valutare la tecnologia di un componente specifico. Il benchmarking viene spesso utilizzato all'inizio di un nuovo progetto, quando non è ancora disponibile una soluzione completa da sottoporre a test di carico.

Puoi creare i tuoi test di benchmarking personalizzati oppure utilizzare test standard del settore, come TPC-DS, per il benchmark dei carichi di lavoro. I benchmark di settore sono utili quando devi confrontare ambienti diversi. Quelli personalizzati, invece, sono indicati per analizzare tipi specifici di operazioni che prevedi di eseguire nell'architettura.

In fase di benchmarking, è importante effettuare delle operazioni preliminari sull'ambiente di test al fine di garantire la validità dei risultati. Dovrai eseguire lo stesso benchmark più volte, per verificare di avere acquisito ogni eventuale variazione nel corso del tempo.

Dal momento che, di solito, l'esecuzione dei benchmark è più rapida di quella dei test di carico, il benchmarking può essere utilizzato sin dalle prime fasi della pipeline di implementazione, così da fornire al team feedback più rapidi sulle deviazioni delle prestazioni. Quando valuti un cambiamento significativo in un componente o servizio, i benchmark possono essere un modo rapido per verificare se l'impegno necessario per apportare la modifica sia giustificato. L'utilizzo del benchmarking in combinazione con i test di carico è importante perché questi ultimi forniscono indicazioni sulle prestazioni del carico di lavoro in fase di produzione.

Passaggi dell'implementazione

- Pianifica e definisci:
  - Definisci gli obiettivi, la baseline, gli scenari di test, le metriche, ad esempio l'utilizzo della CPU, la latenza o il throughput, e i KPI per il tuo benchmark.
  - Concentrati sui requisiti degli utenti in termini di esperienza utente e su fattori come i tempi di risposta e l'accessibilità.
  - Individua uno strumento di benchmark adatto al tuo carico di lavoro. Puoi utilizzare i servizi AWS (come Amazon CloudWatch) o uno strumento di terze parti compatibile con il tuo carico di lavoro.
- Configura ed esegui l'instrumentazione:
  - Imposta il tuo ambiente e configura le risorse.
  - Implementa il monitoraggio e la creazione di log per acquisire i risultati dei test.
- Esegui i test di benchmark e monitora:
  - Esegui i test di benchmark e monitora i parametri durante il test.
- Analizza e documenta:
  - Documenta il processo di benchmark e gli esiti.
  - Analizza i risultati per identificare i colli di bottiglia, le tendenze e le aree di miglioramento.
  - Usa i risultati dei test per prendere decisioni sull'architettura e modificare il carico di lavoro. Questa operazione può includere la modifica dei servizi o l'adozione di nuove funzionalità.

- Ottimizza e ripeti:
  - Modifica le configurazioni e le allocazioni delle risorse in base ai tuoi benchmark.
  - Ripeti il test del carico di lavoro dopo i cambiamenti per convalidare i miglioramenti.
  - Documenta le informazioni e ripeti il processo per identificare altre aree di miglioramento.

Risorse

Documenti correlati:

- AWS Architecture Center
- AWS Partner Network
- Biblioteca di soluzioni di AWS
- Centro conoscenze di AWS
- Amazon CloudWatch RUM
- Amazon CloudWatch Synthetics
- Genomics workflows, Part 5: automated benchmarking
- Benchmark and optimize endpoint deployment in Amazon SageMaker JumpStart

Video correlati:

- AWS re:Invent 2023 - Benchmarking AWS Lambda cold starts
- Benchmarking stateful services in the cloud
- This is my Architecture
- Optimize applications through Amazon CloudWatch RUM
- Demo of Amazon CloudWatch Synthetics

Esempi correlati:

- AWS Esempi di
- Esempi di SDK AWS
- Test del carico distribuito
- Misurazione dei tempi di caricamento delle pagine con Amazon CloudWatch Synthetics
- Client Web Amazon CloudWatch RUM

PERF01-BP07 Uso di un approccio basato sui dati per le scelte dell'architettura

Definisci un approccio chiaro e basato sui dati per le scelte dell'architettura e verificare che vengano utilizzati i servizi e le configurazioni cloud corretti per soddisfare le tue esigenze aziendali specifiche.

Anti-pattern comuni:

- Ritieni che l'architettura corrente diventi statica e non venga aggiornata nel corso del tempo.
- Le tue scelte dell'architettura si basano su ipotesi e supposizioni.
- Introduci modifiche all'architettura nel tempo senza giustificazioni.

Vantaggi dell'adozione di questa best practice: con un approccio ben definito per le scelte dell'architettura, utilizzi i dati per influenzare la progettazione del carico di lavoro e prendere decisioni informate nel tempo.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Affidati all'esperienza e alle competenze interne in materia di cloud o utilizza risorse esterne, come casi d'uso pubblicati o whitepaper, per scegliere risorse e servizi per la tua architettura. È necessario definire con cura un processo che incoraggi la sperimentazione e il benchmarking con i servizi che possono essere utilizzati nel carico di lavoro.

I backlog dei carichi di lavoro critici devono consistere non solo in storie che offrono funzionalità rilevanti per l'azienda e gli utenti, ma anche in storie tecniche che definiscono la presentazione dell'architettura per il carico di lavoro. Questa presentazione include i nuovi progressi tecnologici e i nuovi servizi e li adotta sulla base di dati e giustificazioni adeguate. Verifica che l'architettura sia a prova di futuro e non diventi obsoleta.

Passaggi dell'implementazione

- Interagisci con le principali parti interessate per definire i requisiti del carico di lavoro, comprese le prestazioni, la disponibilità e le considerazioni sui costi. Includi fattori quali il numero di utenti e il modello di utilizzo del tuo carico di lavoro.
- Crea una presentazione dell'architettura o un backlog tecnologico a cui venga assegnata la priorità insieme al backlog funzionale.
- Valuta e identifica i diversi servizi cloud (per ulteriori dettagli, consulta PERF01-BP01 Informazioni e identificazione dei servizi e delle funzionalità cloud disponibili).

- Esplora i diversi modelli di architettura, come microservizi o serverless, che soddisfano i tuoi requisiti di prestazioni (per maggiori dettagli, consulta PERF01-BP02 Utilizzo delle indicazioni del provider cloud o di un partner appropriato per conoscere gli schemi di architettura e le best practice).

- Consulta altri team, diagrammi architetturali e risorse, come AWS Solution Architect, il Centro di architettura AWS e AWS Partner Network, per scegliere l'architettura più adatta al tuo carico di lavoro.

- Definisci i parametri, come il throughput e il tempo di risposta, che possono aiutarti a valutare le prestazioni del tuo carico di lavoro.
- Sperimenta e utilizza i parametri definiti per convalidare le prestazioni dell'architettura selezionata.
- Monitora continuamente e apporta le modifiche necessarie per mantenere ottimali le prestazioni della tua architettura.
- Documenta l'architettura e le decisioni selezionate come riferimento per aggiornamenti e apprendimenti futuri.
- Rivedi e aggiorna continuamente l'approccio di selezione dell'architettura in base agli apprendimenti, alle nuove tecnologie e ai parametri che indicano un problema o un cambiamento necessario nell'approccio attuale.

Risorse

Documenti correlati:

- Biblioteca di soluzioni di AWS
- Centro conoscenze di AWS
- Architectural Patterns to Build End-to-End Data Driven Applications on AWS

Video correlati:

- This is my Architecture
- AWS re:Invent 2021 - Data-driven enterprise: Going from vision to value
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2023 - Optimize cost and performance and track progress toward mitigation

- AWS re:Invent 2022 - AWS optimization: Actionable steps for immediate results

Esempi correlati:

- AWS Esempi di
- Esempi di SDK AWS

Calcolo e hardware

Questions

- PERF 2. In che modo selezioni e utilizzi le risorse di elaborazione nel tuo carico di lavoro?

PERF 2. In che modo selezioni e utilizzi le risorse di elaborazione nel tuo carico di lavoro?

La soluzione ottimale in termini di calcolo per un determinato carico di lavoro potrebbe variare in base alla progettazione dell'applicazione, ai modelli di utilizzo e alle impostazioni di configurazione. Le architetture possono utilizzare diverse soluzioni di calcolo per vari componenti e impiegare funzionalità diverse per migliorare le prestazioni. Selezionare la soluzione di calcolo sbagliata per un'architettura può ridurre l'efficienza delle prestazioni.

Best practice

- PERF02-BP01 Selezione delle migliori opzioni di elaborazione per il carico di lavoro
- PERF02-BP02 Identificazione delle funzionalità e configurazione di calcolo disponibili
- PERF02-BP03 Raccogli metriche relative al calcolo
- PERF02-BP04 Configurazione e dimensionamento corretto delle risorse di elaborazione
- PERF02-BP05 Dimensionamento dinamico delle risorse di elaborazione
- PERF02-BP06 Uso di acceleratori di elaborazione ottimizzati basati su hardware

PERF02-BP01 Selezione delle migliori opzioni di elaborazione per il carico di lavoro

La selezione dell'opzione di elaborazione più appropriata per il carico di lavoro consente di migliorare le prestazioni, ridurre i costi non necessari dell'infrastruttura e diminuire le attività operative richieste per mantenere il carico di lavoro.

Anti-pattern comuni:

- Si utilizza la stessa opzione di elaborazione utilizzata on-premises.
- Non si conoscono le opzioni, le funzionalità e le soluzioni di cloud computing e come queste migliorino le prestazioni di elaborazione.
- Si effettua il provisioning eccessivo dell'opzione di elaborazione per soddisfare i requisiti di dimensionamento o prestazioni, quando il passaggio a una nuova opzione di elaborazione soddisferebbe le caratteristiche del carico di lavoro in modo più preciso.

Vantaggi dell'adozione di questa best practice: identificando i requisiti di elaborazione e valutando le opzioni disponibili è possibile rendere il carico di lavoro più efficiente in termini di risorse.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Per ottimizzare i carichi di lavoro cloud e ottenere prestazioni efficienti, è importante selezionare le opzioni di elaborazione più appropriate per il tuo caso d'uso e i requisiti di prestazioni. AWS offre una varietà di opzioni di elaborazione che soddisfano diversi carichi di lavoro nel cloud. Ad esempio, è possibile utilizzare Amazon EC2 per avviare e gestire server virtuali, AWS Lambda per eseguire codice senza dover allocare o gestire server, Amazon ECS o Amazon EKS per eseguire e gestire container o AWS Batch per elaborare grandi volumi di dati in parallelo. In base alle tue esigenze di dimensionamento ed elaborazione, scegli e configura la soluzione di elaborazione ottimale per la tua situazione. Puoi anche prendere in considerazione l'utilizzo di più tipi di soluzioni di elaborazione in un unico carico di lavoro in quanto ognuna ha i suoi vantaggi e svantaggi.

I passaggi seguenti ti guidano nella selezione delle opzioni di elaborazione giuste per soddisfare le caratteristiche del carico di lavoro e i requisiti prestazionali.

Passaggi dell'implementazione

- Comprendi i requisiti di elaborazione del tuo carico di lavoro. I requisiti essenziali da considerare includono le esigenze di elaborazione, gli schemi di traffico, gli schemi di accesso ai dati, le esigenze di dimensionamento e i requisiti di latenza.
- Scopri i vari servizi di elaborazione AWS per il tuo carico di lavoro. Per ulteriori informazioni, consulta PERF01-BP01 Informazioni e identificazione dei servizi e delle funzionalità cloud disponibili. Ecco alcune importanti opzioni di elaborazione AWS, le caratteristiche e i casi d'uso più comuni:

AWSServizio	Caratteristiche chiave	Casi di utilizzo comune
Amazon Elastic Compute Cloud (Amazon EC2)	Dispone di un'opzione dedicata per hardware, requisiti di licenza, ampia selezione di diverse famiglie di istanze, tipi di processori e acceleratori di elaborazione	Migrazioni con rehosting (lift and shift), applicazione monolitica, ambienti ibridi, applicazioni aziendali
Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS)	Implementazione semplice, ambienti coerenti, scalabile	Microservizi, ambienti ibridi
AWS Lambda	Servizio di elaborazione serverless che esegue il codice in risposta agli eventi e gestisce automaticamente le risorse di elaborazione sottostanti.	Microservizi, applicazioni basate su eventi
AWS Batch	Procede ad allocare e scalare in modo efficient e e dinamico le risorse di elaborazione di di Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS e AWS Fargate, con la possibilità di utilizzare istanze spot o on-demand in base ai requisiti del tuo lavoro	HPC, addestramento dei modelli di ML

AWSServizio	Caratteristiche chiave	Casi di utilizzo comune
Amazon Lightsail	Applicazione Linux e Windows preconfigurata per l'esecuzione di piccoli carichi di lavoro	Applicazioni Web semplici, sito Web personalizzato

- Valuta i costi (come la tariffa oraria o il trasferimento dei dati) e il sovraccarico di gestione (come l'applicazione di patch e il dimensionamento) associati a ciascuna opzione di elaborazione.
- Esegui esperimenti e benchmarking in un ambiente non di produzione per identificare quale opzione di elaborazione può soddisfare al meglio i requisiti del tuo carico di lavoro.
- Dopo aver sperimentato e identificato la tua nuova soluzione di calcolo, pianifica la migrazione e convalida i parametri prestazionali.
- Utilizza gli strumenti di monitoraggio AWS come Amazon CloudWatch e i servizi di ottimizzazione come AWS Compute Optimizer per ottimizzare continuamente le risorse di elaborazione in base a modelli di utilizzo reali.

Risorse

Documenti correlati:

- Elaborazione in cloud con AWS
- Tipi di istanza di Amazon EC
- Amazon EKS Containers: Amazon EKS Worker Nodes
- Amazon ECS Containers: Amazon ECS Container Instances
- Funzioni: configurazione della funzione Lambda
- Prescriptive Guidance for Containers
- Prescriptive Guidance for Serverless

Video correlati:

- AWS re:Invent 2023 - AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 - New Amazon Elastic Compute Cloud generative AI capabilities in AMS
- AWS re:Invent 2023 - What's new with Amazon Elastic Compute Cloud

- AWS re:Invent 2023 - Smart savings: Amazon Elastic Compute Cloud cost-optimization strategies
- AWS re:Invent 2021 - Powering next-gen Amazon Elastic Compute Cloud: Deep dive on the Nitro System
- AWS re:Invent 2019 - Optimize performance and cost for your AWS compute
- AWS re:Invent 2019 - Amazon Elastic Compute Cloud foundations
- AWS re:Invent 2022 - Deploy ML models for inference at high performance and low cost
- AWS re:Invent 2019 - Optimize performance and cost for your AWS compute
- Amazon EC2 foundations
- Deploy ML models for inference at high performance and low cost

Esempi correlati:

- Migrating the Web application to containers
- Esecuzione di un "Hello, World!" serverless
- Workshop su Amazon EKS
- Workshop su Amazon EC2
- Efficient and Resilient Workloads with Amazon Elastic Compute Cloud Auto Scaling
- Migrating to AWS Graviton with Container Services

PERF02-BP02 Identificazione delle funzionalità e configurazione di calcolo disponibili

Comprendi le opzioni e le funzionalità di configurazione disponibili per il tuo servizio di calcolo in modo da fornire la giusta quantità di risorse e migliorare l'efficienza delle prestazioni.

Anti-pattern comuni:

- Non valuti le opzioni di calcolo o le famiglie di istanze disponibili rispetto alle caratteristiche del carico di lavoro.
- Esegui il provisioning eccessivo delle risorse di calcolo per soddisfare i requisiti di picco della domanda.

Vantaggi dell'adozione di questa best practice: acquisisci familiarità con le funzionalità e le configurazioni di calcolo di AWS in modo da poter utilizzare una soluzione di calcolo ottimizzata per soddisfare le caratteristiche e le esigenze del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Ogni soluzione di calcolo ha disponibili configurazioni e funzionalità specifiche per supportare caratteristiche e requisiti diversi del carico di lavoro. Scopri in che modo puoi completare al meglio il tuo carico di lavoro e quali opzioni di configurazione sono le migliori per la tua applicazione. Esempi di tali opzioni includono la famiglia di istanze, le dimensioni, le caratteristiche (GPU, I/O), il bursting, i timeout, le dimensioni delle funzioni, le istanze di container e la simultaneità. Se per il carico di lavoro è stata utilizzata la stessa opzione di calcolo per oltre quattro settimane e sai già che le caratteristiche resteranno uguali in futuro, puoi utilizzare AWS Compute Optimizer per scoprire se la tua attuale opzione di calcolo è adatta ai carichi di lavoro dal punto di vista della CPU e della memoria.

Passaggi dell'implementazione

- Comprendi i requisiti del carico di lavoro, come CPU, memoria e latenza.
- Consulta la documentazione e le best practice AWS per scoprire le opzioni di configurazione consigliate che possono contribuire a migliorare le prestazioni di calcolo. Ecco alcune opzioni di configurazione chiave da considerare:

Opzione di configurazione	Esempi
Tipo di istanza	• Le istanze ottimizzate per il calcolo sono l'ideale per i carichi di lavoro che richiedono un rapporto vCPU/memoria molto elevato. • Le istanze ottimizzate per la memoria offrono grandi quantità di memoria per carichi di lavoro intensivi in questo senso. • Le istanze ottimizzate per l'archiviazione sono progettate per carichi di lavoro che richiedono un accesso frequente e sequenziale in lettura e scrittura (IOPS) all'archiviazione locale.
Modello tariffario	• Le istanza on demand ti consentono di utilizzare la capacità di calcolo su base oraria o al secondo, senza impegni a lungo

Opzione di configurazione	Esempi
	termine e sono ideali per il bursting oltre le esigenze di base per le prestazioni. • Savings Plans offrono risparmi significativi rispetto alle istanze on demand in cambio dell'impegno a utilizzare una quantità specifica di potenza di elaborazione per un periodo di uno o tre anni. • Le istanze spot ti consentono di sfruttare la capacità inutilizzata delle istanze con uno sconto per i carichi di lavoro stateless e tolleranti ai guasti.
Auto Scaling	Usa la configurazione Auto Scaling per abbinare le risorse di calcolo ai modelli di traffico.
Dimensionamento	• Usa Compute Optimizer per ricevere un efficace suggerimento di machine learning riguardo alla configurazione più adatta alle tue caratteristiche di elaborazione. • Usa AWS Lambda Power Tuning per selezionare la configurazione migliore per la tua funzione Lambda.
Acceleratori di calcolo basati su hardware	• Le istanza a calcolo accelerato eseguono funzioni come l'elaborazione grafica o la corrispondenza di schemi di dati in modo più efficiente rispetto alle alternative basate sulla CPU. • Per i carichi di lavoro di machine learning, sfrutta l'hardware specifico per il tuo carico di lavoro, come AWS Trainium, AWS Inferentia e Amazon EC2 DL1.

Risorse

Documenti correlati:

- Elaborazione in cloud con AWS
- Tipi di istanza di Amazon EC
- Processor State Control for Your Amazon EC2 Instance
- Amazon EKS Containers: Amazon EKS Worker Nodes
- Amazon ECS Containers: Amazon ECS Container Instances
- Funzioni: configurazione della funzione Lambda

Video correlati:

- AWS re:Invent 2023 – AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 – New Amazon EC2 generative AI capabilities in Console di gestione AWS
- AWS re:Invent 2023 – What's new with Amazon EC2
- AWS re:Invent 2023 – Smart savings: Amazon EC2 cost-optimization strategies
- AWS re:Invent 2021 – Powering next-gen Amazon EC2: Deep dive on the Nitro System
- AWS re:Invent 2019 – Amazon EC2 foundations
- AWS re:Invent 2022 – Optimizing Amazon EKS for performance and cost on AWS

Esempi correlati:

- Codice dimostrativo di Compute Optimizer
- Workshop relativo alle istanze spot Amazon EC2
- Efficient and Resilient Workloads with Amazon EC2 AWS Auto Scaling
- Workshop per sviluppatori Graviton
- AWS for Microsoft workloads immersion day
- AWS for Linux workloads immersion day
- Codice dimostrativo AWS Compute Optimizer
- Workshop su Amazon EKS

PERF02-BP03 Raccogli metriche relative al calcolo

Registra e monitora i parametri relativi all'elaborazione per comprendere meglio le prestazioni delle tue risorse di elaborazione e migliorarne le prestazioni e l'utilizzo.

Anti-pattern comuni:

- Utilizzi solo i file di log manuali per la ricerca dei parametri.
- Utilizzi solo i parametri predefiniti registrati dal software di monitoraggio.
- Revisione dei parametri solo quando c'è un problema.

Vantaggi dell'adozione di questa best practice: la raccolta dei parametri relativi alle prestazioni ti aiuta ad allineare le prestazioni delle applicazioni ai requisiti aziendali per garantire il rispetto delle esigenze dei carichi di lavoro. Può anche aiutarti a migliorare costantemente le prestazioni e l'utilizzo delle risorse del tuo carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I carichi di lavoro del cloud possono generare grandi volumi di dati quali parametri, log ed eventi. Nel Cloud AWS, la raccolta delle metriche è un passaggio fondamentale per migliorare la sicurezza, l'efficienza dei costi, le prestazioni e la sostenibilità. AWS fornisce un'ampia gamma di metriche relative alle prestazioni utilizzando servizi di monitoraggio come Amazon CloudWatch per fornirti informazioni preziose. Metriche come CPU l'utilizzo, l'utilizzo della memoria, l'I/O del disco e la rete in entrata e in uscita possono fornire informazioni sui livelli di utilizzo o sui colli di bottiglia delle prestazioni. Utilizza tali parametri come parte di un approccio basato sui dati per ottimizzare e ottimizzare le risorse del tuo carico di lavoro. L'ideale sarebbe raccogliere tutti i parametri relativi alle tue risorse di elaborazione in un'unica piattaforma con policy di conservazione implementate per supportare costi e obiettivi operativi.

Passaggi dell'implementazione

- Identifica quali parametri relativi alle prestazioni sono rilevanti per il tuo carico di lavoro. Raccogli i parametri sull'utilizzo delle risorse e sul modo in cui opera il tuo carico di lavoro nel cloud (come il tempo di risposta e il throughput).
  - Metriche EC2 predefinite di Amazon
  - Metriche ECS predefinite di Amazon
  - Metriche EKS predefinite di Amazon

  - Parametri predefiniti di Lambda
  - Parametri EC2 della memoria e del disco di Amazon
- Scegli e configura la soluzione di registrazione e monitoraggio giusta per il tuo carico di lavoro.
  - AWS native Observability
  - AWS Distro per OpenTelemetry
  - Amazon Managed Service per Prometheus
- Definisci il filtro e l'aggregazione richiesti per i parametri in base ai requisiti del tuo carico di lavoro.
  - Quantifica i parametri delle applicazioni personalizzate con Amazon CloudWatch Logs e filtri metrici
  - Raccogli metriche personalizzate con il tagging CloudWatch strategico di Amazon
- Configura le policy di conservazione dei dati per i parametri in modo che corrispondano ai tuoi obiettivi operativi e di sicurezza.
  - Conservazione dei dati predefinita per le metriche CloudWatch
  - Conservazione dei dati predefinita per i registri CloudWatch
- Se necessario, crea allarmi e notifiche per i parametri in modo da rispondere in modo proattivo ai problemi relativi alle prestazioni.
  - Crea allarmi per metriche personalizzate utilizzando il rilevamento delle anomalie di Amazon CloudWatch
  - Crea metriche e allarmi per pagine Web specifiche con Amazon CloudWatch RUM
- Usa l'automazione per implementare gli agenti di aggregazione di parametri e log.
  - AWS Systems Manager automazione
  - OpenTelemetryCollezionista

Risorse

Documenti correlati:

- Monitoraggio e osservabilità
- Migliori pratiche: implementazione dell'osservabilità con AWS
- CloudWatch Documentazione Amazon
- Raccogli metriche e log EC2 dalle istanze Amazon e dai server locali con l'agente CloudWatch
- Accesso ad Amazon CloudWatch Logs per AWS Lambda

- Utilizzo dei CloudWatch log con istanze di container
- Publish custom metrics
- AWS Answers: Centralized Logging
- AWS Servizi che pubblicano metriche CloudWatch
- Monitoraggio di Amazon EKS su AWS Fargate

Video correlati:

- AWS re:Invent 2023 — [LAUNCH] Monitoraggio delle applicazioni per carichi di lavoro moderni
- AWS re:Invent 2023 — Implementazione dell'osservabilità delle applicazioni
- AWS re:Invent 2023 — Creazione di una strategia di osservabilità efficace
- AWS re:Invent 2023 — Osservabilità senza interruzioni con Distro per AWS OpenTelemetry
- Gestione delle prestazioni delle applicazioni su AWS

Esempi correlati:

- AWS per Linux Workload Immersion Day- Amazon CloudWatch
- Monitoraggio di ECS cluster e container Amazon
- Monitoraggio con CloudWatch dashboard Amazon
- EKSWorkshop Amazon

PERF02-BP04 Configurazione e dimensionamento corretto delle risorse di elaborazione

Configura e dimensiona correttamente le risorse di elaborazione per soddisfare i requisiti di prestazioni del carico di lavoro ed evitare un utilizzo insufficiente o eccessivo delle risorse.

Anti-pattern comuni:

- Ignori i requisiti di prestazioni del carico di lavoro, con il risultato del provisioning eccessivo o insufficiente delle risorse di elaborazione.
- Scegli semplicemente l’istanza più grande o più piccola disponibile per tutti i carichi di lavoro.
- Usi una sola famiglia di istanze per semplificare la gestione.
- Ignori i suggerimenti di AWS Cost Explorer o Compute Optimizer per il corretto dimensionamento.
- Non rivaluti il carico di lavoro in base all’idoneità dei nuovi tipi di istanza.

- Certifichi solo un numero limitato di configurazioni di istanza per l'organizzazione.

Vantaggi dell'adozione di questa best practice il corretto dimensionamento delle risorse di elaborazione garantisce un funzionamento ottimale nel cloud evitando il provisioning eccessivo o insufficiente delle risorse. Il corretto dimensionamento delle risorse di elaborazione comporta in genere prestazioni ottimali e una migliore esperienza cliente, riducendo al contempo i costi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Il dimensionamento corretto consente alle organizzazioni di gestire la propria infrastruttura cloud in modo efficiente ed economico, rispettando al contempo le esigenze aziendali. Il provisioning eccessivo di risorse cloud può tradursi in costi aggiuntivi, mentre il provisioning insufficiente può comportare prestazioni non soddisfacenti e un'esperienza negativa per il cliente. AWS offre strumenti come AWS Compute Optimizer e AWS Trusted Advisor che sfruttano dati cronologici per fornire consigli sul corretto dimensionamento delle risorse di elaborazione.

Passaggi dell'implementazione

- Scegli il tipo di istanza più adatto alle tue esigenze:
  - Come faccio a scegliere il tipo di istanza Amazon EC2 appropriata per il mio carico di lavoro?
  - Selezione del tipo di istanza basata su attributi per Amazon EC2 Fleet
  - Create an Auto Scaling group using attribute-based instance type selection
  - Optimizing your Kubernetes compute costs with Karpenter consolidation
- Analizza le varie caratteristiche di prestazione del tuo carico di lavoro e come queste sono correlate a memoria, rete e utilizzo della CPU. Utilizza questi dati per scegliere le risorse che meglio corrispondono al profilo del tuo carico di lavoro e agli obiettivi di prestazioni.
- Monitora l'utilizzo delle risorse con gli strumenti di monitoraggio di AWS come Amazon CloudWatch.
- Seleziona la configurazione corretta per la risorsa di elaborazione.
  - Per carichi di lavoro effimeri, valuta i parametri dell'istanza di Amazon CloudWatch, ad esempio `CPUUtilization`, per identificare se l'istanza è sovra o sottoutilizzata.
  - Per i carichi di lavoro stabili, esegui i controlli con gli strumenti di ridimensionamento corretto di AWS, come AWS Compute Optimizer e AWS Trusted Advisor a intervalli regolari per individuare le opportunità di ottimizzazione e ridimensionamento corretto della risorsa di elaborazione.

- Esegui il test delle modifiche apportate alla configurazione in un ambiente non di produzione prima di implementarle in un ambiente live.
- Rivaluta costantemente nuove offerte di elaborazione e confrontale con le esigenze del carico di lavoro.

Risorse

Documenti correlati:

- Elaborazione in cloud con AWS
- Tipi di istanza di Amazon EC
- Container Amazon ECS: istanze di container di Amazon ECS
- Amazon EKS Container: nodi worker di Amazon EKS
- Funzioni: configurazione della funzione Lambda
- Controllo degli stati del processore dell'istanza Amazon EC2

Video correlati:

- Amazon EC2 foundations
- AWS re:Invent 2023 – AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 – New Amazon EC2 generative AI capabilities in Console di gestione AWS
- AWS re:Invent 2023 – What's new with Amazon EC2
- AWS re:Invent 2023 – Smart savings: Amazon EC2 cost-optimization strategies
- AWS re:Invent 2021 – Powering next-gen Amazon EC2: Deep dive on the Nitro System
- AWS re:Invent 2019 – Amazon EC2 foundations

Esempi correlati:

- Codice dimostrativo AWS Compute Optimizer
- Workshop su Amazon EKS
- Right-sizing recommendations

PERF02-BP05 Dimensionamento dinamico delle risorse di elaborazione

Sfrutta l'elasticità del cloud per scalare dinamicamente le risorse di elaborazione per soddisfare le tue esigenze ed evitare un provisioning eccessivo o insufficiente per il tuo carico di lavoro.

Anti-pattern comuni:

- Risposta agli allarmi aumentando manualmente la capacità.
- Utilizzi le stesse linee guida per il dimensionamento (generalmente infrastruttura statica) di quelle on-premises.
- Dopo un evento di dimensionamento, lasci una capacità aumentata anziché ridurre il dimensionamento.

Vantaggi dell'adozione di questa best practice: la configurazione e il test dell'elasticità delle risorse di elaborazione possono aiutarti a risparmiare denaro, mantenere i benchmark delle prestazioni e migliorare l'affidabilità al variare del traffico.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

AWS offre la flessibilità necessaria per scalare le risorse in modo dinamico attraverso una varietà di meccanismi di dimensionamento per soddisfare le variazioni della domanda. In combinazione con i parametri relativi all'elaborazione, il dimensionamento dinamico consente ai carichi di lavoro di rispondere automaticamente alle modifiche e utilizzare il set ottimale di risorse di elaborazione per raggiungere l'obiettivo.

Puoi adottare varie strategie di approccio per associare l'offerta di risorse alla domanda.

- Approccio al tracciamento degli obiettivi: monitora il parametro di dimensionamento e aumenta o diminuisci automaticamente la capacità in base alle esigenze.
- Dimensionamento predittivo: procedi a ridurre orizzontalmente in previsione delle tendenze giornaliere e settimanali.
- Approccio basato sulla pianificazione: imposta il tuo programma di dimensionamento in base alle variazioni di carico prevedibili.
- Scalabilità del servizio: scegli i servizi (come quelli serverless) che si dimensionano automaticamente per progettazione.

Assicurati che le implementazioni dei carichi di lavoro siano in grado di gestire eventi che prevedono l'aumentare verticalmente e il ridurre verticalmente.

Passaggi dell'implementazione

- Istanze di elaborazione, container e funzioni forniscono tutti meccanismi di elasticità, in combinazione con il dimensionamento automatico o sotto forma di funzionalità del servizio. Ecco alcuni esempi di meccanismi di dimensionamento automatico:

Meccanismo di scalabilità automatica	Dove usarlo
Amazon EC2 Auto Scaling	Assicura di disporre del numero corretto di istanze Amazon EC2 disponibili per gestire il carico dell'applicazione.
Application Auto Scaling	Dimensiona in automatico le risorse per singoli servizi AWS oltre Amazon EC2, ad esempio, funzioni AWS Lambda o servizi Amazon Elastic Container Service (Amazon ECS).
Kubernetes Cluster Autoscaler/Karpenter	Dimensiona automaticamente i cluster Kubernetes.

- Si parla spesso di dimensionamento con servizi di calcolo come le istanze Amazon EC2 o le funzioni AWS Lambda. Assicurati di considerare anche la configurazione di servizi non di calcolo come AWS Glue per soddisfare la domanda.
- Verifica che i parametri per il dimensionamento corrispondano alle caratteristiche del carico di lavoro da implementare. Se implementi un'applicazione di transcodifica video, è previsto il 100% di utilizzo della CPU e non deve essere il parametro principale. Utilizza la profondità della coda dei processi di transcodifica. Se necessario, puoi utilizzare una metrica personalizzata per la tua policy di dimensionamento. Per scegliere la metrica corretta, consulta le linee guida seguenti per Amazon EC2:
  - La metrica deve essere una metrica di utilizzo valida e descrivere il livello di impiego di un'istanza.
  - Il valore del parametro deve aumentare e diminuire in proporzione al numero di istanze nel gruppo con scalabilità automatica.

- Assicurati di utilizzare il dimensionamento dinamico anziché il dimensionamento manuale per il tuo gruppo Auto Scaling. È consigliabile utilizzare le policy di dimensionamento del monitoraggio degli obiettivi nel dimensionamento dinamico
- Verifica che le implementazioni dei carichi di lavoro siano in grado di gestire entrambi gli eventi di dimensionamento (aumento e riduzione). Ad esempio, puoi usare la cronologia delle attività per verificare le attività di ridimensionamento per un gruppo Auto Scaling.
- Analizza il tuo carico di lavoro per individuare modelli prevedibili e dimensionare le tue risorse in modo proattivo, anticipando variazioni nella domanda previste e pianificate. Con il dimensionamento predittivo puoi eliminare la necessità di offrire capacità in eccedenza. Per ulteriori informazioni, consulta Dimensionamento predittivo con Amazon EC2 Auto Scaling.

Risorse

Documenti correlati:

- Elaborazione in cloud con AWS
- Tipi di istanza di Amazon EC
- Container Amazon ECS: istanze di container di Amazon ECS
- Amazon EKS Container: nodi worker di Amazon EKS
- Funzioni: configurazione della funzione Lambda
- Controllo degli stati del processore dell'istanza Amazon EC2
- Deep Dive on Amazon ECS Cluster Auto Scaling
- Introducing Karpenter – An Open-Source High-Performance Kubernetes Cluster Autoscaler

Video correlati:

- AWS re:Invent 2023 – AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 – New Amazon EC2 generative AI capabilities in AWS Management Console
- AWS re:Invent 2023 – What's new with Amazon EC2
- AWS re:Invent 2023 – Smart savings: Amazon EC2 cost-optimization strategies
- AWS re:Invent 2021 – Powering next-gen Amazon EC2: Deep dive on the Nitro System
- AWS re:Invent 2019 – Amazon EC2 foundations

Esempi correlati:

- Esempi di gruppo di Amazon EC2 Auto Scaling
- Workshop su Amazon EKS
- Scale your Amazon EKS workloads by running on IPv6

PERF02-BP06 Uso di acceleratori di elaborazione ottimizzati basati su hardware

Usa gli acceleratori hardware per eseguire determinate funzioni in modo più efficiente rispetto alle alternative basate sulla CPU.

Anti-pattern comuni:

- Nel carico di lavoro non hai confrontato un'istanza per uso generico con un'istanza dedicata in grado di offrire prestazioni più elevate e costi inferiori.
- Usi gli acceleratori di calcolo basati su hardware per attività in cui sono più efficienti le alternative basate su CPU.
- Utilizzo delle GPU non monitorato.

Vantaggi dell'adozione di questa best practice: utilizzando gli acceleratori basati su hardware, come le unità di elaborazione grafica (GPU) e le serie di porte programmabili sul campo (FPGA), è possibile eseguire determinate funzioni di elaborazione in modo più efficiente.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Le istanze a calcolo accelerato forniscono l'accesso agli acceleratori di calcolo basati su hardware, come GPU e FPGA. Questi acceleratori hardware eseguono alcune funzioni, come l'elaborazione grafica o la rilevazione della corrispondenza dei modelli di dati, in modo più efficiente rispetto alle alternative basate su CPU. Molti carichi di lavoro accelerati, come il rendering grafico, la transcodifica e il machine learning, sono altamente variabili in termini di utilizzo di risorse. Esegui questo hardware solo per il tempo necessario e disattivalo con l'automazione quando non serve per migliorare l'efficienza complessiva delle prestazioni.

Passaggi dell'implementazione

- Identifica le istanza a calcolo accelerato in grado di soddisfare i tuoi requisiti.
- Per i carichi di lavoro di machine learning, sfrutta l'hardware specifico per il tuo carico di lavoro, come AWS Trainium, AWS Inferentia e Amazon EC2 DL1. AWS Le istanze Inferentia come le

istanze Inf2 offrono fino al 50% in più di prestazioni per watt rispetto alle istanze Amazon EC2 paragonabili.

- Raccogli i parametri di utilizzo delle istanze a calcolo accelerato. Ad esempio, puoi utilizzare l'agente CloudWatch per acquisire metriche quali `utilization_gpu` e `utilization_memory` per le tue GPU, come illustrato in Collect NVIDIA GPU metrics with Amazon CloudWatch.
- Ottimizza il codice, il funzionamento della rete e le impostazioni degli acceleratori hardware per garantire il pieno utilizzo dell'hardware sottostante.
  - Ottimizza le impostazioni GPU
  - Monitoraggio e ottimizzazione delle GPU nell'AMI per il deep learning
  - Optimizing I/O for GPU performance tuning of deep learning training in Amazon SageMaker
- Utilizza le librerie e i driver per GPU più recenti e performanti.
- Utilizza l'automazione per rilasciare le istanze GPU non in uso.

Risorse

Documenti correlati:

- Utilizzo di GPU su Amazon Elastic Container Service
- Istanze GPU
- Istanze con AWS Trainium
- Istanze con AWS Inferentia
- Let's Architect! Architecting with custom chips and accelerators

- Calcolo accelerato
- Amazon EC2 VT1 Instances
- Come faccio a scegliere il tipo di istanza Amazon EC2 appropriata per il mio carico di lavoro?
- Choose the best AI accelerator and model compilation for computer vision inference with Amazon SageMaker

Video correlati:

- AWS re:Invent 2021 - How to select Amazon Elastic Compute Cloud GPU instances for deep learning

- AWS re:Invent 2022 - [NEW LAUNCH!] Introducing AWS Inferentia2-based Amazon EC2 Inf2 instances
- AWS re:Invent 2022 - Accelerate deep learning and innovate faster with AWS Trainium
- AWS re:Invent 2022 - Deep learning on AWS with NVIDIA: From training to deployment

Esempi correlati:

- Amazon SageMaker and NVIDIA GPU Cloud (NGC)
- Use SageMaker with Trainium and Inferentia for optimized deep learning training and inferencing workloads
- Optimizing NLP models with Amazon Elastic Compute Cloud Inf1 instances in Amazon SageMaker

Gestione dei dati

Questions

- PERF 3. In che modo archivi, gestisci e accedi ai dati nel tuo carico di lavoro?

PERF 3. In che modo archivi, gestisci e accedi ai dati nel tuo carico di lavoro?

La soluzione ottimale per la gestione dei dati in un sistema specifico varia in base al tipo di dati (blocco, file o oggetto), agli schemi di accesso (casuali o sequenziali), al throughput necessario, alla frequenza di accesso (online, offline, archivio), alla frequenza di aggiornamento (WORM, dinamico) e ai vincoli di disponibilità e durata. I carichi di lavoro Well-Architected utilizzano archivi dati appositamente progettati che impiegano diverse funzionalità per migliorare le prestazioni.

Best practice

- PERF03-BP01 Uso di un archivio dati dedicato che supporta al meglio i requisiti di accesso e archiviazione dei dati
- PERF03-BP02 Valutazione delle opzioni di configurazione disponibili per datastore
- PERF03-BP03 Raccolta e registrazione dei parametri delle prestazioni del datastore
- PERF03-BP04 Implementazione di strategie per migliorare le prestazioni delle query nel datastore
- PERF03-BP05 Implementazione di modelli di accesso ai dati che utilizzano la memorizzazione nella cache

PERF03-BP01 Uso di un archivio dati dedicato che supporta al meglio i requisiti di accesso e archiviazione dei dati

Comprendi le caratteristiche dei dati (come la condivisione, le dimensioni, la dimensione della cache, gli schemi di accesso, la latenza, il throughput e la persistenza dei dati) per selezionare i data store (archiviazione o database) dedicati per il tuo carico di lavoro.

Anti-pattern comuni:

- Continui a utilizzare un datastore per via dell'esperienza e delle competenze interne relative a quel particolare tipo di soluzione di database.
- Ritieni che tutti i carichi di lavoro abbiano requisiti di accesso e archiviazione di dati simili.
- Non hai implementato un catalogo di dati per eseguire l'inventario dei tuoi asset.

Vantaggi dell'adozione di questa best practice: la comprensione delle caratteristiche e dei requisiti dei dati ti consente di determinare la tecnologia di archiviazione più efficiente e performante appropriata per le tue esigenze del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Quando selezioni e implementi l'archiviazione di dati, assicurati che le caratteristiche di query, dimensionamento e archiviazione supportino i requisiti dei dati del carico di lavoro. AWS fornisce numerose tecnologie di database e archiviazione di dati, tra cui archiviazione a blocchi, archiviazione di oggetti, archiviazione di streaming, file system, database di libro mastro, relazionali, chiave-valore, di documenti, in memoria, a grafo, di serie temporali. Ogni soluzione di gestione dei dati offre soluzioni e configurazioni adatte a gestire i tuoi casi d'uso e modelli di dati. Comprendendo le caratteristiche e i requisiti dei dati, puoi abbandonare la tecnologia di archiviazione monolitica e gli approcci restrittivi e validi per tutti, per concentrarti sulla gestione dei dati in modo appropriato.

Passaggi dell'implementazione

- Esegui un inventario dei vari tipi di dati esistenti nel tuo carico di lavoro.
- Comprendi e documenta le caratteristiche e i requisiti dei dati, tra cui:
  - Tipo di dati (non strutturati, semi-strutturati, relazionali)
  - Volume e crescita dei dati
  - Durabilità dei dati: persistenti, effimeri, transitori

- Requisiti ACID (atomicità, coerenza, isolamento, durabilità)
- Schemi di accesso ai dati (con uso intensivo di lettura o scrittura)
- Latenza
- Throughput
- IOPS (operazioni di input/output al secondo)
- Periodo di conservazione dei dati

- Scopri i diversi archivi di dati (servizi di archiviazione e database) disponibili per il carico di lavoro in AWS che possono soddisfare le caratteristiche dei tuoi dati (come illustrato in PERF01-BP01 Informazioni e identificazione dei servizi e delle funzionalità cloud disponibili). Alcuni esempi di tecnologie di archiviazione AWS e delle loro caratteristiche chiave sono:

Tipo	Services	Caratteristiche chiave
Archiviazione di oggetti	Amazon S3	Scalabilità illimitata, alta disponibilità e molteplici opzioni di accessibilità. L'accesso a oggetti e il relativo trasferimento da e verso Amazon S3 può utilizzare un servizio, come Transfer Acceleration o Punti di accesso, per supportare la posizione, le esigenze di sicurezza e i modelli di accesso.
Archiviazione	Amazon Glacier	Progettato per l'archiviazione dei dati.
Archiviazione in streaming	Amazon Kinesis Streaming gestito da Amazon per Apache Kafka (Amazon MSK)	Acquisizione e archiviazione efficienti dei dati in streaming.

Tipo	Services	Caratteristiche chiave
File system condiviso	Amazon Elastic File System (Amazon EFS)	File system montabile a cui è possibile accedere da più tipi di soluzioni di calcolo.
File system condiviso	Amazon FSx	Sviluppato con le più recenti soluzioni di calcolo AWS per supportare i 4 file system più comunemente utilizzati: NetApp ONTAP, OpenZFS, Windows File Server e Lustre. Latenza, throughput e IOPS di Amazon FSx variano a seconda del file system; è necessario considerare attentamente questi elementi quando si deve selezionare il file system in modo conforme ai requisiti dei carichi di lavoro.
Storage a blocchi	Amazon Elastic Block Store (Amazon EBS)	Servizio di storage a blocchi scalabile e a elevate prestazioni progettato per Amazon Elastic Compute Cloud (Amazon EC2). Amazon EBS include storage su SSD per carichi di lavoro transazionali e intensivi dal punto di vista dell'IOPS, oltre a storage su HDD per carichi di lavoro con throughput intensivo.

Tipo	Services	Caratteristiche chiave
Database relazionale	Amazon Aurora, Amazon RDS, Amazon Redshift.	Progettati per supportare le transazioni ACID (atomicità, coerenza, isolamento, durabilità) e per mantenere l'integrità referenziale e una solida coerenza dei dati. Molte applicazioni tradizionali, Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) ed e-commerce utilizzano database relazionali per archiviare i propri dati.
Database chiave-valore	Amazon DynamoDB	Ottimizzato per schemi di accesso di uso comune, in genere per archiviare e recuperare grandi volumi di dati. Le app Web dal traffico elevato, i sistemi di e-commerce e le applicazioni di videogiochi sono casi d'uso tipici dei database chiave-valore.
Database di documenti	Amazon DocumentDB	Progettato per archiviare dati semistrutturati come documenti simili a JSON. Questi database aiutano gli sviluppatori a creare e aggiornare rapidamente applicazioni quali gestione di contenuti, cataloghi e profili utente.

Tipo	Services	Caratteristiche chiave
Database in memoria	Amazon ElastiCache, Amazon MemoryDB per Redis	Vengono utilizzati per applicazioni che richiedon o accesso in tempo reale ai dati, bassissima latenza ed elevatissimo throughput. È possibile utilizzare database in memoria per la memorizza zione nella cache delle applicazioni, la gestione delle sessioni, la classifica dei giochi, l'archivio delle caratteri stiche ML a bassa latenza, il sistema di messaggistica dei microservizi e un meccanism o di streaming a elevato throughput.
Database a grafo	Amazon Neptune	Utilizzato con le applicazi oni che devono navigare ed eseguire query su milioni di relazioni tra set di dati a grafo altamente connessi, con una latenza misurata in milliseco ndi su larga scala. Molte aziende utilizzano database a grafo per il rilevamen to di attività fraudolente, i social network e i motori di raccomandazione.

Tipo	Services	Caratteristiche chiave
Database di serie temporali	Amazon Timestream	Utilizzato per raccogliere, sintetizzare e derivare in modo efficiente approfond imenti dai dati che cambiano nel tempo. I database di serie temporali sono spesso utilizzati dalle applicazioni IoT, DevOps e dalla telemetri a industriale.
Colonna ampia	Amazon Keyspaces (per Apache Cassandra)	Utilizza tabelle, righe e colonne, ma a differenza di un database relaziona le, i nomi e il formato delle colonne possono variare da riga a riga all'interno della stessa tabella. In genere, gli store colonnari sono utilizzati nelle applicazioni industriali su larga scala per la manutenzione delle apparecchiature, la gestione delle flotte e l'ottimizzazione dei percorsi.

Tipo	Services	Caratteristiche chiave
Di libri mastri	Database Amazon Quantum Ledger (Amazon QLDB)	Fornisce un'autorità centralizzata e affidabile per mantenere un registro delle transazioni scalabile , immutabile e verificabile tramite crittografia per ogni applicazione. I database di libri mastri vengono utilizzati per sistemi di record, catena di fornitura, registrazioni e persino transazioni bancarie.

- Per una piattaforma dati, sfrutta l'architettura dei dati moderna di AWS per integrare data lake, data warehouse e archivi dati appositamente progettati.
- Le domande chiave da porsi quando si sceglie un data store per il carico di lavoro sono le seguenti:

Domanda	Aspetti da considerare
Come sono strutturati i dati?	• Se i dati non sono strutturati, prendi in considerazione un archivio di oggetti, come Amazon S3, o un database NoSQL, come Amazon DocumentDB • Per i dati di tipo chiave-valore, valuta DynamoDB, Amazon ElastiCache (Redis OSS) o Amazon MemoryDB
Quale livello di integrità referenziale è richiesto ?	• Per i vincoli di chiave esterna, i database relazionali come Amazon RDS e Aurora possono fornire livello di integrità richiesto. • In genere, in un modello di dati NoSQL, i dati vengono denormalizzati in un singolo documento o in una raccolta di documenti da recuperare in un'unica richiesta, anziché essere uniti tra diversi documenti o tabelle.

Domanda	Aspetti da considerare
È richiesta la conformità ACID (atomicità, coerenza, isolamento, durabilità)?	• Se sono necessarie proprietà ACID associate ai database relazionali, valuta un database relazionale come Amazon RDS e Aurora. • Se è necessaria un'elevata coerenza per i database NoSQL, puoi utilizzare le elevate consistenza di lettura con DynamoDB.
Come cambierà nel tempo l'archiviazione? In che modo questo avrà effetto sulla scalabilità?	• I database serverless, come DynamoDB e i Database Amazon Quantum Ledger (Amazon QLDB) offrono la scalabilità dinamica. • Per i database relazionali sono previsti limiti massimi per l'archiviazione allocata, al raggiungimento dei quali si rende spesso necessario partizionare orizzontalmente tali database tramite meccanismi quali la partizione.
Qual è la proporzione di query in lettura rispetto alle quelle in scrittura? Il caching potrebbe probabilmente migliorare le prestazio ni?	• Per i carichi di lavoro gravosi in termini di lettura, può essere utile un livello di memorizzazione nella cache, come ElastiCache o DAX, se il database è DynamoDB • È anche possibile passare le operazioni di lettura alle repliche di lettura con database relazionali come Amazon RDS.

Domanda	Aspetti da considerare
Hanno priorità più elevata le operazioni di archiviazione e modifica OLTP, Online Transaction Processing) o quelle di recupero e report (OLAP - Online Analytical Processing)?	• Per un'elaborazione transazionale letta così com'è a elevato throughput, prendi in considerazione un database NoSQL come DynamoDB. • Per schemi di lettura complessi con throughput elevato (come il join) con un uso coerente di Amazon RDS. • Per le query analitiche, prendi in considera zione un database a colonne, come Amazon Redshift, o l'esportazione dei dati su Amazon S3, nonché l'esecuzione di analisi mediante Athena o Amazon QuickSight.
Che livello di durabilità è necessario per i dati?	• Aurora replica automaticamente i dati su tre zone di disponibilità all'interno di una regione, il che significa che i dati sono altamente durevoli con minori probabilità di perdite. • DynamoDB viene automaticamente replicato in più zone di disponibilità per offrire livelli elevati di disponibilità e durabilit à dei dati. • Amazon S3 offre il 99,999999999 di durabilità. Molti servizi di database, come Amazon RDS e DynamoDB, supportano l'esportazione di dati su Amazon S3 per la conservazione e l'archiviazione a lungo termine.

Domanda	Aspetti da considerare
È presente il desiderio di abbandonare i motori di database commerciali o i costi di licenza?	• Valuta motori open-source come PostgreSQL e MySQL su Amazon RDS o Aurora. • Sfrutta AWS Database Migration Service e AWS Schema Conversion Tool per eseguire le migrazioni dai motori di database commerciali a quelli open-source
Quali sono le aspettative operative per il database? Il passaggio ai servizi gestiti è una priorità?	• Utilizzare Amazon RDS, invece di Amazon EC2, e scegliere DynamoDB o Amazon DocumentDB, invece di ospitare in autonomia un database NoSQL, riduce le spese operative.
Come avviene attualmente l'accesso al database? È solo un accesso da applicazione o sono presenti utenti Business Intelligence (BI) e altre applicazioni pronte all'uso connesse?	• In presenza di dipendenze da strumenti esterni, potresti dover mantenere la compatibilità con i database che supportano. Amazon RDS è del tutto compatibile con le varie versioni dei motori che supporta, tra cui Microsoft SQL Server, Oracle, MySQL e PostgreSQL.

- Esegui esperimenti e benchmarking in un ambiente non di produzione per identificare quale datastore può soddisfare al meglio i requisiti del tuo carico di lavoro.

Risorse

Documenti correlati:

- Tipi di volume Amazon EBS
- Archiviazione Amazon EC2
- Amazon EFS: Amazon EFS Performance
- Amazon FSx for Lustre Performance
- Amazon FSx for Windows File Server Performance
- Amazon Glacier: documentazione di Amazon Glacier
- Amazon S3: considerazioni su velocità e prestazioni delle richieste

- Archiviazione nel cloud in AWS
- Amazon EBS I/O Characteristics
- Database su cloud con AWS
- AWS Database Caching
- DynamoDB Accelerator
- Best practice di Amazon Aurora
- Prestazioni di Amazon RedShift
- Amazon Athena top 10 performance tips
- Amazon Redshift Spectrum best practices
- Amazon DynamoDB best practices
- Choose between Amazon EC2 and Amazon RDS
- Best practice per l'implementazione di Amazon ElastiCache

Video correlati:

- AWS re:Invent 2023: Improve Amazon Elastic Block Store efficiency and be more cost-efficient
- AWS re:Invent 2023: Optimizing storage price and performance with Amazon Simple Storage Service
- AWS re:Invent 2023: Building and optimizing a data lake on Amazon Simple Storage Service
- AWS re:Invent 2022: Building modern data architectures on AWS
- AWS re:Invent 2022: Building data mesh architectures on AWS
- AWS re:Invent 2023: Deep dive into Amazon Aurora and its innovations
- AWS re:Invent 2023: Advanced data modeling with Amazon DynamoDB
- AWS re:Invent 2022: Modernize apps with purpose-built databases
- Amazon DynamoDB deep dive: Advanced design patterns

Esempi correlati:

- AWS Purpose Built Databases Workshop
- Databases for Developers
- AWS Modern Data Architecture Immersion Day

- Build a Data Mesh on AWS
- Amazon S3 Examples
- Optimize Data Pattern using Amazon Redshift Data Sharing
- Migrazioni dei database
- MS SQL Server - AWS Database Migration Service (AWS DMS) Replication Demo
- Database Modernization Hands On Workshop
- Esempi di Amazon Neptune

PERF03-BP02 Valutazione delle opzioni di configurazione disponibili per datastore

Comprendi e valuta le varie funzionalità e opzioni di configurazione disponibili per i tuoi datastore per ottimizzare lo spazio di archiviazione e le prestazioni per il tuo carico di lavoro.

Anti-pattern comuni:

- Utilizzi un solo tipo di storage, ad esempio Amazon EBS, per tutti i carichi di lavoro.
- Utilizzi la capacità di IOPS allocata per tutti i carichi di lavoro senza test reali su tutti i livelli di archiviazione.
- Non conosci le opzioni di configurazione della soluzione di gestione dei dati scelta.
- Ti basi soltanto sull'aumento delle dimensioni dell'istanza, senza tenere conto di altre opzioni di configurazione disponibili.
- Non esegui il test delle caratteristiche di dimensionamento del tuo datastore.

Vantaggi dell'adozione di questa best practice: esplorare le configurazioni del datastore e sperimentare con esse può consentire di ridurre il costo dell'infrastruttura, migliorare le prestazioni e ridurre l'impegno richiesto per mantenere i carichi di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Un carico di lavoro può utilizzare uno o più datastore in base ai requisiti di archiviazione di dati e relativo accesso. Per ottimizzare prestazioni, efficienza e costi, è necessario valutare gli schemi di accesso ai dati per determinare le configurazioni appropriate del datastore. Nella valutazione delle opzioni di datastore, prendi in considerazione vari aspetti come le opzioni di archiviazione, la memoria, l'elaborazione, la replica di lettura, i requisiti di coerenza, il pool di connessioni e le opzioni

di caching. Esegui esperimenti con queste diverse opzioni di configurazione per migliorare i parametri di efficienza delle prestazioni.

Passaggi dell'implementazione

- Esamina le configurazioni correnti (come il tipo di istanza, la dimensione di archiviazione o la versione del motore di database) del tuo datastore.
- Consulta documentazione e best practice AWS per scoprire le opzioni di configurazione consigliate che possono contribuire a migliorare le prestazioni del datastore. Le principali opzioni da considerare per il datastore sono le seguenti:

Opzione di configurazione	Esempi
Riduzione del carico delle letture (come le repliche di lettura e la memorizzazione nella cache)	• Per le tabelle DynamoDB, è possibile eliminare il carico delle letture grazie a DAX per la memorizzazione nella cache. • Puoi creare un cluster Amazon ElastiCache (Redis OSS) e configurare l'applicazione in modo che legga prima dalla cache e quindi passi al database se l'elemento richiesto non è presente. • I database relazionali come Amazon RDS e Aurora, nonché i database NoSQL allocati, come Neptune e Amazon DocumentDB, supportano tutti l'aggiunta di repliche di lettura per eliminare il carico creato dalle parti di lettura nel carico di lavoro. • I database serverless come DynamoDB si dimensionano automaticamente. Assicurat i di avere abbastanza unità di capacità di lettura (RCU) allocate per gestire il carico di lavoro.
Dimensionamento delle scritture (come la partizione delle chiavi di partizione o l'introdu zione di una coda)	• Per i database relazionali, è possibile aumentare la dimensione dell'istanza per gestire un maggiore carico di lavoro o aumentare la capacità di IOPS allocata per

Opzione di configurazione	Esempi
	gestire un maggior throughput verso l'archivi azione sottostante. • È anche possibile introdurre una coda davanti al database, invece di eseguire direttamente la scrittura su di esso. Questo schema consente di disaccoppiare l'acquisi zione dal database e controllare il flusso, in modo che il database sia in grado di gestirlo. • Raggruppare in batch le richieste di scrittura , anziché creare molte transazioni di breve durata, può aiutare a migliorare il throughpu t in database relazionali con un elevato volume in scrittura. • I database serverless come DynamoDB possono dimensionare automaticamente il throughput in scrittura oppure è possibile regolare le unità di capacità in scrittura (WCU) allocate, a seconda della modalità di capacità. • È tuttavia possibile che si verifichino problemi con le partizioni hot quando si raggiungono i limiti di throughput per una determinata chiave di partizione. Questo problema può essere arginato scegliendo una chiave di partizione con una distribuz ione più uniforme o eseguendo lo sharding in lettura della chiave di partizione.

Opzione di configurazione	Esempi
Policy per gestire il ciclo di vita dei set di dati	• Puoi utilizzare il ciclo di vita Amazon S3 per gestire gli oggetti durante il loro ciclo di vita. In caso di schemi di accesso sconosciu ti, mutevoli o imprevedibili, puoi utilizzar e il Piano intelligente Amazon S3, che monitora gli schemi di accesso e sposta in automatico gli oggetti che non hanno fatto registrare accessi a livelli di accessi più economici. Sfrutta i parametri di Amazon S3 Storage Lens per individuare opportunità di ottimizzazione e lacune nella gestione del ciclo di vita. • La gestione del ciclo di vita di Amazon EFS gestisce automaticamente l'archivi azione di file a costi contenuti per i file system.
Gestione e pooling delle connessioni	• È possibile utilizzare Server proxy per Amazon RDS con Amazon RDS e Aurora per gestire le connessioni al database. • I database serverless come DynamoDB non hanno connessioni associate, ma valuta la capacità assegnata e le policy di dimension amento automatico per affrontare i picchi nel carico.

- Esegui esperimenti e benchmarking in un ambiente non di produzione per identificare quale opzione di configurazione può soddisfare i requisiti del tuo carico di lavoro.
- Dopo gli esperimenti, pianifica la migrazione e convalida i parametri delle prestazioni.
- Usa strumenti di monitoraggio AWS (come Amazon CloudWatch) e ottimizzazione (come Amazon S3 Storage Lens) per ottimizzare continuamente il tuo datastore utilizzando schemi di utilizzo reali.

Risorse

Documenti correlati:

- Archiviazione nel cloud in AWS
- Tipi di volume Amazon EBS
- Archiviazione Amazon EC
- Amazon EFS: Amazon EFS Performance
- Amazon FSx for Lustre Performance
- Amazon FSx for Windows File Server Performance
- Amazon Glacier: documentazione di Amazon Glacier
- Amazon S3: considerazioni su velocità e prestazioni delle richieste
- Amazon EBS I/O Characteristics
- Database su cloud con AWS
- AWS Database Caching
- DynamoDB Accelerator
- Best practice di Amazon Aurora
- Prestazioni di Amazon RedShift
- Amazon Athena top 10 performance tips
- Amazon Redshift Spectrum best practices
- Amazon DynamoDB best practices

Video correlati:

- AWS re:Invent 2023: Improve Amazon Elastic Block Store efficiency and be more cost-efficient
- AWS re:Invent 2023: Optimize storage price and performance with Amazon Simple Storage Service
- AWS re:Invent 2023: Building and optimizing a data lake on Amazon Simple Storage Service
- AWS re:Invent 2023: What's new with AWS file storage
- AWS re:Invent 2023: Dive deep into Amazon DynamoDB

Esempi correlati:

- AWS Purpose Built Databases Workshop
- Databases for Developers
- AWS Modern Data Architecture Immersion Day

- Dimensionamento automatico di Amazon EBS
- Amazon S3 Examples
- Esempi di Amazon DynamoDB
- Esempi di migrazione di database con AWS
- Workshop sulla modernizzazione dei database
- Working with parameters on your Amazon RDS for Postgress DB

PERF03-BP03 Raccolta e registrazione dei parametri delle prestazioni del datastore

Tieni traccia e registra i parametri delle prestazioni pertinenti per il tuo datastore per capire l'andamento delle prestazioni delle soluzioni di gestione dei dati. Questi parametri possono aiutarti a ottimizzare il tuo datastore, verificare che i requisiti del carico di lavoro siano rispettati e fornire una panoramica chiara sull'andamento delle prestazioni del carico di lavoro.

Anti-pattern comuni:

- Utilizzi solo i file di log manuali per la ricerca dei parametri.
- Pubblichi i parametri solo sugli strumenti interni utilizzati dal tuo team e non hai un quadro completo del carico di lavoro.
- Utilizzo solo dei parametri predefiniti registrati dal software di monitoraggio selezionato.
- Revisione dei parametri solo quando c'è un problema.
- Monitori solo i parametri a livello di sistema, senza acquisire i parametri di accesso ai dati o di utilizzo.

Vantaggi dell'adozione di questa best practice: la definizione di una linea di base delle prestazioni ti aiuta a comprendere il comportamento normale e i requisiti dei carichi di lavoro. Gli schemi anomali possono essere identificati ed eliminati più rapidamente, per migliorare le prestazioni e l'affidabilità del datastore.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Per monitorare le prestazioni dei datastore, devi registrare più parametri delle prestazioni in un periodo di tempo. Ciò consente di rilevare le anomalie e di misurare le prestazioni rispetto ai parametri aziendali, per verificare che le esigenze del carico di lavoro siano rispettate.

I parametri devono includere sia il sistema sottostante che supporta il datastore sia i parametri del database. I parametri del sistema sottostante possono includere utilizzo della CPU, memoria, spazio di archiviazione su disco disponibile, I/O su disco, percentuale di riscontri nella cache e parametri di rete in entrata e in uscita, mentre i parametri del datastore possono includere transazioni al secondo, query principali, velocità media delle query, tempi di risposta, utilizzo degli indici, blocco delle tabelle, timeout delle query e numero di connessioni aperte. Questi dati sono cruciali per capire l'andamento del carico di lavoro e come viene utilizzata la soluzione di gestione dei dati. Utilizza tali parametri come parte di un approccio basato sui dati per mettere a punto e ottimizzare le risorse del tuo carico di lavoro.

Utilizza strumenti, librerie e sistemi che registrano misure delle prestazioni relative alle prestazioni del database.

Passaggi dell'implementazione

- Determina i principali parametri delle prestazioni da monitorare per il tuo datastore.
  - Parametri e dimensioni di Amazon S3
  - Monitoraggio di parametri in un'istanza Amazon RDS
  - Monitoring DB load with Performance Insights on Amazon RDS
  - Panoramica sul monitoraggio avanzato
  - DynamoDB Metrics and dimensions
  - Monitoraggio di DynamoDB Accelerator
  - Monitoring Amazon MemoryDB with Amazon CloudWatch
  - Quali parametri è opportuno monitorare?
  - Monitoring Amazon Redshift cluster performance
  - Timestream metrics and dimensions
  - Amazon CloudWatch metrics for Amazon Aurora
  - Creazione di log e monitoraggio in Amazon Keyspaces (per Apache Cassandra)
  - Monitoring Amazon Neptune Resources
- Utilizza una soluzione di registrazione e monitoraggio approvata per raccogliere queste metriche. Amazon CloudWatch può raccogliere i parametri per tutte le risorse dell'architettura. Puoi anche raccogliere e pubblicare parametri personalizzati per ottenere parametri aziendali o derivati. Utilizza CloudWatch o soluzioni di terze parti per impostare allarmi che indicano quando le soglie vengono superate.

- Verifica se il monitoraggio dei datastore può trarre vantaggio da una soluzione di machine learning che rileva le anomalie delle prestazioni.
  - Amazon DevOps Guru per Amazon RDS offre visibilità sui problemi di prestazioni e fornisce suggerimenti per le azioni correttive.
- Configura la conservazione dei dati nella soluzione di monitoraggio e registrazione per soddisfare i tuoi obiettivi operativi e di sicurezza.
  - Conservazione dei dati predefinita per i parametri CloudWatch
  - Conservazione dei dati predefinita per i parametri CloudWatch Logs

Risorse

Documenti correlati:

- AWS Database Caching
- Amazon Athena top 10 performance tips
- Amazon Aurora best practices
- DynamoDB Accelerator
- Amazon DynamoDB best practices
- Amazon Redshift Spectrum best practices
- Prestazioni di Amazon RedShift
- Database su cloud AWS
- Approfondimenti sulle prestazioni di Amazon RDS

Video correlati:

- AWS re:Invent 2022 - Performance monitoring with Amazon RDS and Aurora, featuring Autodesk
- Database Performance Monitoring and Tuning with Amazon DevOps Guru for Amazon RDS
- AWS re:Invent 2023 - What's new with AWS file storage
- AWS re:Invent 2023 - Dive deep into Amazon DynamoDB
- AWS re:Invent 2023 - Building and optimizing a data lake on Amazon S3
- AWS re:Invent 2023 - What's new with AWS file storage
- AWS re:Invent 2023 - Dive deep into Amazon DynamoDB

- Best Practices for Monitoring Redis Workloads on Amazon ElastiCache

Esempi correlati:

- Framework di raccolta dei parametri di ingestione del set di dati AWS
- Workshop relativo al monitoraggio di Amazon RDS
- AWS Purpose Built Databases Workshop

PERF03-BP04 Implementazione di strategie per migliorare le prestazioni delle query nel datastore

Implementa le strategie per ottimizzare i dati e migliorare le query sui dati in modo da consentire una maggiore scalabilità e prestazioni più efficienti per il tuo carico di lavoro.

Anti-pattern comuni:

- Non suddividi i dati in partizioni nel tuo datastore.
- I dati vengono archiviati in un solo formato di file nel tuo datastore.
- Non usi gli indici nel tuo datastore.

Vantaggi dell'adozione di questa best practice: l'ottimizzazione delle prestazioni dei dati e delle query si traduce in maggiore efficienza, costi inferiori e migliore esperienza utente.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

L'ottimizzazione di dati e query è un aspetto critico dell'efficienza delle prestazioni in un datastore, poiché influisce sulle prestazioni e sulla reattività dell'intero carico di lavoro cloud. Le query non ottimizzate possono comportare un maggiore utilizzo delle risorse e rallentamenti, riducendo così l'efficienza complessiva di un datastore.

L'ottimizzazione dei dati include diverse tecniche per garantire prestazioni efficienti per l'archiviazione di dati e il relativo accesso. Ciò aiuta anche a migliorare le prestazioni delle query in un datastore. Le strategie chiave includono il partizionamento, la compressione e la denormalizzazione dei dati, che contribuiscono a ottimizzare i dati sia per l'archiviazione che per l'accesso.

Passaggi dell'implementazione

- Esamina e analizza le query sui dati critiche che vengono eseguite nel tuo datastore.

- Individua le query lente del tuo datastore e utilizza i piani di query per comprenderne lo stato attuale.
  - Analisi del piano di query in Amazon Redshift
  - Using EXPLAIN and EXPLAIN ANALYZE in Athena
- Implementa le strategie per migliorare le prestazioni delle query. Ecco alcune strategie chiave:
  - Utilizzo di un formato di file colonnare (come Parquet o ORC).
  - Compressione dei dati nel datastore per ridurre lo spazio di archiviazione e il funzionamento di I/O.
  - Partizionamento dei dati per suddividere i dati in parti più piccole e ridurre i tempi di analisi dei dati.
    - Partizionamento dei dati in Athena
    - Partitions and data distribution
  - Indicizzazione dei dati sulle colonne comuni della query.
  - Uso delle viste materializzate per le domande frequenti.
    - Understanding materialized views
    - Creating materialized views in Amazon Redshift
  - Scelta dell'operazione di unione corretta per la query. Quando unisci due tabelle, specifica la tabella più grande sul lato sinistro dell'unione e la tabella più piccola sul lato destro.
  - Miglioramento della latenza e riduzione del numero di operazioni di I/O del database grazie alla soluzione di cache distribuita.
  - Manutenzione regolare, ad esempio vacuum, reindicizzazione ed esecuzione di statistiche.
- La sperimentazione e i test delle strategie in un ambiente non di produzione.

Risorse

Documenti correlati:

- Best practice di Amazon Aurora
- Prestazioni di Amazon RedShift
- Amazon Athena top 10 performance tips
- AWS Database Caching
- Best practice per l'implementazione di Amazon ElastiCache
- Partizionamento dei dati in Athena

Video correlati:

- AWS re:Invent 2023 - AWS storage cost-optimization best practices
- AWS re:Invent 2022 - Performance monitoring with Amazon RDS and Aurora, featuring Autodesk
- Optimize Amazon Athena Queries with New Query Analysis Tools

Esempi correlati:

- AWS Purpose Built Databases Workshop

PERF03-BP05 Implementazione di modelli di accesso ai dati che utilizzano la memorizzazione nella cache

Implementa modelli di accesso che possano trarre vantaggio dalla memorizzazione dei dati nella cache per il recupero rapido dei dati a cui si accede di frequente.

Anti-pattern comuni:

- Memorizzare nella cache dati che cambiano in maniera frequente.
- Fare affidamento sui dati memorizzati nella cache come se fossero archiviati in modo duraturo e sempre disponibili.
- Non tenere conto della coerenza dei dati memorizzati nella cache.
- Non monitorare l'efficienza dell'implementazione della cache.

Vantaggi dell'adozione di questa best practice: l'archiviazione dei dati in una cache può migliorare la latenza di lettura, il throughput, l'esperienza utente e l'efficienza complessiva, oltre a ridurre i costi.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Una cache è un componente software o hardware progettato per archiviare dati in modo che le richieste future degli stessi dati possano essere soddisfatte più velocemente o in modo più efficiente. I dati memorizzati in una cache possono essere ricostruiti in caso di perdita, ripetendo un calcolo precedente o recuperandolo da un altro datastore.

La memorizzazione dei dati nella cache può essere una delle strategie più efficaci per migliorare le prestazioni complessive delle applicazioni e ridurre il carico sulle origini dati primarie sottostanti.

È possibile memorizzare i dati nella cache a più livelli dell'applicazione, ad esempio all'interno dell'applicazione che effettua chiamate remote, operazione nota come memorizzazione nella cache lato client, o mediante un servizio secondario veloce per l'archiviazione dei dati, operazione nota come memorizzazione nella cache remota.

Memorizzazione nella cache lato client

Con la memorizzazione nella cache lato client, ogni client (un'applicazione o un servizio che interroga il datastore di backend) può archiviare localmente i risultati delle proprie query uniche per un periodo di tempo specificato. Ciò può ridurre il numero di richieste a un datastore attraverso la rete perché viene controllata prima la cache del client locale. Se questa non contiene risultati, l'applicazione può interrogare il datastore e archiviare tali risultati localmente. Questo modello consente a ciascun client di archiviare i dati nella sede più vicina possibile (il client stesso), garantendo così la latenza più bassa possibile. I client possono inoltre continuare a eseguire query quando il datastore di backend non è disponibile, aumentando la disponibilità dell'intero sistema.

Uno svantaggio di questo approccio è che quando sono coinvolti più client, potrebbero archiviare localmente gli stessi dati memorizzati nella cache. Ciò si traduce in un utilizzo duplicato dell'archiviazione e nell'incoerenza dei dati tra questi client. Può accadere che un client memorizzi nella cache i risultati di una query e un minuto dopo un altro client esegua la stessa query ottenendo un risultato diverso.

Memorizzazione nella cache remota

Come soluzione al problema della duplicazione dei dati tra client, è possibile utilizzare un servizio esterno veloce o la memorizzazione nella cache remota per archiviare i dati sottoposti a query. Anziché controllare un datastore locale, ogni client controllerà la cache remota prima di interrogare il datastore di backend. Questa strategia consente di ottenere risposte più coerenti tra i client, una migliore efficienza dei dati archiviati e un volume maggiore di dati memorizzati nella cache, perché lo spazio di archiviazione si dimensiona in maniera indipendente dai client.

Lo svantaggio di una cache remota è che l'intero sistema può registrare una latenza più elevata, poiché è necessario un hop di rete aggiuntivo per controllare la cache remota. Per migliorare la latenza, è possibile utilizzare la memorizzazione nella cache lato client insieme alla memorizzazione nella cache remota, eseguendo così una memorizzazione nella cache su più livelli.

Passaggi dell'implementazione

- Identifica database, API e servizi di rete che potrebbero trarre vantaggio dalla memorizzazione nella cache. I candidati migliori per la memorizzazione nella cache sono i servizi che presentano

carichi di lavoro di lettura elevati, un rapporto lettura/scrittura elevato o che sono costosi da dimensionare.

  - Database Caching
  - Abilitazione della memorizzazione nella cache dell'API per migliorare la velocità di risposta
- Identifica il tipo di strategia di memorizzazione nella cache più adatto al tuo modello di accesso.
  - Caching strategies
  - AWS Caching Solutions
- Attieniti alle best practice sulla memorizzazione nella cache per il tuo archivio dati.
- Configura una strategia di invalidazione della cache per tutti i dati, ad esempio un TTL (Time-to-live), che permetta di bilanciare attualità dei dati e riduzione della pressione sul datastore di backend.
- Abilita funzionalità quali tentativi di connessione automatici, backoff esponenziale, timeout lato client e pool di connessioni nel client, se disponibili, che possono migliorare prestazioni e affidabilità.
  - Best practices: Redis clients and Amazon ElastiCache (Redis OSS)
- Monitora la percentuale di riscontri nella cache con un obiettivo dell'80% o superiore. Valori inferiori possono indicare una dimensione della cache insufficiente o un modello di accesso che non sfrutta la memorizzazione nella cache.
  - Which metrics should I monitor?
  - Best practices for monitoring Redis workloads on Amazon ElastiCache
  - Monitoring best practices with Amazon ElastiCache (Redis OSS) using Amazon CloudWatch
- Implementa la replica dei dati per eliminare il carico delle letture per più istanze e migliorare prestazioni e disponibilità della lettura dei dati.

Risorse

Documenti correlati:

- Using the Amazon ElastiCache Well-Architected Lens
- Monitoring best practices with Amazon ElastiCache (Redis OSS) using Amazon CloudWatch
- Quali parametri è opportuno monitorare?
- Performance at Scale with Amazon ElastiCache whitepaper
- Sfide e strategie del caching

Video correlati:

- Amazon ElastiCache Learning Path
- Design for success with Amazon ElastiCache best practices
- AWS re:Invent 2020 - Design for success with Amazon ElastiCache best practices
- AWS re:Invent 2023 - [LAUNCH] Introducing Amazon ElastiCache Serverless
- AWS re:Invent 2022 - 5 great ways to reimagine your data layer with Redis
- AWS re:Invent 2021 - Deep dive on Amazon ElastiCache (Redis OSS)

Esempi correlati:

- Boosting MySQL database performance with Amazon ElastiCache (Redis OSS)

Reti e distribuzione di contenuti

Questions

- PERF 4. In che modo selezioni e configuri le risorse di rete nel carico di lavoro?

PERF 4. In che modo selezioni e configuri le risorse di rete nel carico di lavoro?

La soluzione di rete ottimale per un carico di lavoro varia in base a latenza, requisiti di throughput, jitter e larghezza di banda. I vincoli fisici, ad esempio le risorse utente o on-premises, determinano le opzioni di posizione. Questi vincoli possono essere compensati con le posizioni edge o la collocazione delle risorse.

Best practice

- PERF04-BP01 In che modo la rete influisce sulle prestazioni
- PERF04-BP02 Valuta le funzionalità di rete disponibili
- PERF04-BP03 Scegli la connettività dedicata o la VPN appropriata per il tuo carico di lavoro
- PERF04-BP04 Utilizzo del bilanciamento del carico per distribuire il traffico su più risorse
- PERF04-BP05 Scelta dei protocolli di rete per migliorare le prestazioni
- PERF04-BP06 Scegli la posizione del carico di lavoro in base ai requisiti di rete
- PERF04-BP07 Ottimizzazione della configurazione di rete in base alle metriche

PERF04-BP01 In che modo la rete influisce sulle prestazioni

Analizza e comprendi in che modo le decisioni correlate alla rete influiscono sul carico di lavoro per fornire prestazioni efficienti e una migliore esperienza utente.

Anti-pattern comuni:

- Tutto il traffico passa attraverso i data center esistenti.
- Si instrada tutto il traffico attraverso i firewall centrali anziché utilizzare strumenti di sicurezza di rete nativi del cloud.
- Si effettua il provisioning delle connessioni AWS Direct Connect senza comprendere gli effettivi requisiti di utilizzo.
- Quando si definiscono le soluzioni di rete, non si considerano le caratteristiche del carico di lavoro e l'overhead della crittografia.
- Per le soluzioni di rete nel cloud si utilizzano concetti e strategie on-premises.

Vantaggi dell'adozione di questa best practice: la comprensione dell'impatto della rete sulle prestazioni del carico di lavoro ti aiuta a identificare i potenziali colli di bottiglia, migliorare l'esperienza dell'utente, aumentare l'affidabilità e ridurre la manutenzione operativa al variare del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

La rete è responsabile della connettività tra componenti dell'applicazione, servizi cloud, reti edge e dati on-premises e quindi può avere un forte impatto sulle prestazioni dei carichi di lavoro. Oltre alle prestazioni del carico di lavoro, l'esperienza dell'utente può essere influenzata anche da latenza della rete, larghezza di banda, protocolli, posizione, congestione della rete, jitter, throughput e regole di instradamento.

Predisponi un elenco documentato dei requisiti di rete del carico di lavoro, tra cui latenza, dimensione dei pacchetti, regole di instradamento, protocolli e modelli di traffico di supporto. Esamina le soluzioni di rete disponibili e individua il servizio che soddisfi le caratteristiche di rete del proprio carico di lavoro. Le reti basate sul cloud possono essere ricostruite rapidamente, quindi l'evoluzione dell'architettura di rete nel tempo è necessaria per migliorare l'efficienza delle prestazioni.

Passaggi dell'implementazione:

- Definisci e documenta i requisiti di prestazioni di rete, tra cui metriche come latenza di rete, larghezza di banda, protocolli, posizioni, modelli di traffico (picchi e frequenza), throughput, crittografia, ispezione e regole di instradamento.
- Scopri i principali servizi di rete AWS come VPC, AWS Direct Connect, Elastic Load Balancing (ELB) e Amazon Route 53.
- Acquisisci le seguenti caratteristiche di rete fondamentali:

Caratteristiche	Strumenti e metriche
Caratteristiche fondamentali della rete	• Log di flusso VPC • Log di flusso AWS Transit Gateway • AWS Transit Gateway Parametri di • AWS PrivateLink Parametri di
Caratteristiche di rete dell'applicazione	• Elastic Fabric Adapter • AWS App Mesh Parametri di • Parametri per Gateway Amazon API
Caratteristiche della rete edge	• Parametri di Amazon CloudFront • Parametri di Amazon Route 53 • AWS Global Accelerator Parametri di
Caratteristiche della rete ibrida	• Direct Connect Parametri di • AWS Site-to-Site VPN Parametri di • AWS Client VPN Parametri di • Parametri WAN Cloud AWS
Caratteristiche della sicurezza di rete	• Parametri AWS Shield, AWS WAF e AWS Network Firewall
Caratteristiche del tracciamento	• AWS X-Ray • VPC Reachability Analyzer • Strumento di analisi degli accessi alla rete

Caratteristiche	Strumenti e metriche
	• Amazon Inspector • Amazon CloudWatch RUM

- Esegui il benchmark e testa le prestazioni della rete:
  - Esegui il benchmark del throughput della rete, poiché alcuni fattori possono influire sulle prestazioni della rete Amazon EC2 quando le istanze si trovano nello stesso VPC. Misura la larghezza di banda della rete tra le istanze Amazon EC2 Linux nello stesso VPC.
  - Effettua test di carico per sperimentare soluzioni e opzioni di rete.

Risorse

Documenti correlati:

- Application Load Balancer
- Reti avanzate EC2 su Linux
- Reti avanzate EC2 su Windows
- Gruppi di collocamento EC
- Abilitazione delle reti avanzate con l'Adattatore elastico di rete (ENA) sulle istanze Linux
- Network Load Balancer
- Prodotti di rete con AWS
- Gateway di transito
- Transitioning to latency-based routing in Amazon Route 53
- Endpoint VPC

Video correlati:

- AWS re:Invent 2023 - AWS networking foundations
- AWS re:Invent 2023 - What can networking do for your application?
- AWS re:Invent 2023 - Advanced VPC designs and new capabilities
- AWS re:Invent 2023 - A developer's guide to cloud networking
- AWS re:Invent 2019 - Connectivity to AWS and hybrid AWS network architectures
- AWS re:Invent 2019 - Optimizing Network Performance for Amazon EC2 Instances

- AWS Summit Online - Improve Global Network Performance for Applications
- AWS re:Invent 2020 - Networking best practices and tips with the Well-Architected Framework
- AWS re:Invent 2020 - AWS networking best practices in large-scale migrations

Esempi correlati:

- AWS Transit Gateway and Scalable Security Solutions
- Workshop sulle reti AWS
- Hands-on Network Firewall Workshop
- Observing and Diagnosing your Network on AWS
- Finding and addressing Network Misconfigurations on AWS

PERF04-BP02 Valuta le funzionalità di rete disponibili

Valuta le funzionalità di rete nel cloud che possono aumentare le prestazioni. Misura l'impatto di tali funzionalità attraverso test, parametri e analisi. Ad esempio, sfrutta le funzionalità a livello di rete disponibili per ridurre latenza, distanza di rete o jitter.

Anti-pattern comuni:

- Rimani all'interno di una regione perché è lì che si trova fisicamente la tua sede centrale.
- Utilizzi i firewall anziché i gruppi di sicurezza per filtrare il traffico.
- Interrompi TLS per l'ispezione del traffico anziché affidarti a gruppi di sicurezza, policy degli endpoint e altre funzionalità native del cloud.
- Utilizzi solo la segmentazione basata su sottoreti anziché i gruppi di sicurezza.

Vantaggi dell'adozione di questa best practice la valutazione di tutte le funzionalità e le opzioni del servizio consente di ridurre il costo dell'infrastruttura e l'impegno necessario per mantenere il carico di lavoro e aumentare l'assetto di sicurezza generale. La struttura portante globale di AWS ti aiuta a fornire ai tuoi clienti la migliore esperienza di rete.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

AWS offre servizi come AWS Global Accelerator e Amazon CloudFront per migliorare le prestazioni di rete, mentre la maggior parte dei servizi AWS offre funzionalità di prodotto (come la funzionalità Amazon S3 Transfer Acceleration) per l'ottimizzazione del traffico di rete.

Analizza quali opzioni di configurazione relative alla rete sono disponibili e come possono influire sul tuo carico di lavoro. L'ottimizzazione delle prestazioni dipende dalla comprensione del modo in cui queste opzioni interagiscono con l'architettura e dall'impatto che hanno sulle prestazioni misurate e sull'esperienza utente.

Passaggi dell'implementazione

- Crea l'elenco dei componenti del carico di lavoro.
  - Prendi in considerazione l'uso del WAN di Cloud AWS per creare, gestire e monitorare la rete dell'organizzazione durante la creazione di una rete globale unificata.
  - Monitora le tue reti globali e principali con le metriche di Amazon CloudWatch Logs. Sfrutta Amazon CloudWatch RUM, che fornisce approfondimenti utili per identificare, comprendere e migliorare l'esperienza digitale degli utenti.
  - Visualizza la latenza di rete aggregata tra Regioni AWS e le zone di disponibilità, nonché all'interno di ciascuna zona di disponibilità, sfruttando AWS Network Manager per ottenere informazioni dettagliate sulla relazione fra le prestazioni delle applicazioni e quelle della rete AWS sottostante.
  - Utilizza uno strumento esistente per il database di gestione della configurazione (CMDB) o un servizio come AWS Config per creare un inventario del carico di lavoro e della relativa configurazione.
- Se si tratta di un carico di lavoro esistente, individua e documenta l'analisi di benchmark per le metriche relative alle prestazioni, concentrandoti sui colli di bottiglia e sulle aree da migliorare. Le metriche relative alla rete a livello di prestazioni varieranno a seconda dei requisiti aziendali e delle caratteristiche del carico di lavoro. Come punto di partenza, le seguenti metriche possono essere importanti per la revisione del carico di lavoro: larghezza di banda, latenza, perdita di pacchetti, jitter e ritrasmissioni.
- Se si tratta di un nuovo carico di lavoro, esegui test di carico per individuare i colli di bottiglia delle prestazioni.
- Per tutti i colli di bottiglia di questo tipo individuati, esamina le opzioni di configurazione per le soluzioni in uso per individuare le opportunità di miglioramento delle prestazioni. Consulta le seguenti opzioni e funzionalità di rete fondamentali:

Opportunità di miglioramento	Soluzione
Percorso o instradamenti di rete	Usa lo Strumento di analisi degli accessi alla rete per identificare percorsi o percorsi.
Protocolli di rete	Per informazioni, consultare PERF04-BP05 Scelta dei protocolli di rete per migliorare le prestazioni
Topologia di rete	Valuta i compromessi a livello di operazioni e prestazioni tra VPC Peering e AWS Transit Gateway quando si collegano più account. AWS Transit Gateway semplifica il modo in cui interconnetti tutti i VPC, che possono essere distribuiti su migliaia di Account AWS e in reti on-premises. Condividi AWS Transit Gateway tra più account utilizzando AWS Resource Access Manager. Per informazioni, consultare PERF04-BP 03 Scegli la connettività dedicata o la VPN appropriata per il tuo carico di lavoro

Opportunità di miglioramento	Soluzione
Servizi di rete	AWS Global Accelerator è un servizio di rete che migliora le prestazioni del traffico degli utenti fino al 60% utilizzando l'infrastruttura di rete globale di AWS. Amazon CloudFront può migliorare le prestazioni della distribuzione dei contenuti del tuo carico di lavoro e la latenza a livello globale. Usa Lambda@edge per eseguire funzioni di personalizzazione dei contenuti che CloudFront distribuisce più vicino agli utenti, ridurre la latenza e migliorare le prestazioni. Amazon Route 53 offre opzioni di instradamento basato sulla latenza, instradamento basato sulla geolocalizzazione, instradamento basato sulla geoprossimità e instradamento basato su IP per migliorare le prestazioni del tuo carico di lavoro per un pubblico globale. Rivedi il traffico del carico di lavoro e la posizione dell'utente quando il carico di lavoro è distribuito a livello globale per individuare quale opzione di instradamento è in grado di ottimizzare le prestazioni del carico di lavoro.

Opportunità di miglioramento	Soluzione
Funzionalità delle risorse di archiviazione	Amazon S3 Transfer Acceleration è una funzionalità che consente agli utenti esterni di sfruttare i vantaggi delle ottimizzazioni di rete di CloudFront per il caricamento dei dati in Amazon S3. Ciò migliora le caratteristiche di trasferimento di grandi quantità di dati da posizioni remote prive di connettività dedicata al Cloud AWS. I punti di accesso multi-regione di Amazon S3 rappresentano una funzionalità che replica i contenuti in più regioni e semplifica il carico di lavoro fornendo un punto di accesso. Quando viene utilizzato un punto di accesso multi-reg ione, puoi richiedere o scrivere dati in Amazon S3 con il servizio che identifica il bucket con latenza più bassa.

Opportunità di miglioramento	Soluzione
Funzionalità delle risorse di calcolo	Le interfacce di rete elastica (ENI) utilizzate da istanze Amazon EC2, container e funzioni Lambda sono limitate in base ai flussi. Rivedi i gruppi di collocazione per ottimizzare il throughput di rete di EC2. Per evitare colli di bottiglia a livello di flusso, progetta l'applica zione in modo che utilizzi più flussi. Per monitorare le metriche di rete associate al calcolo e avere maggiore visibilità su di esse, utilizza i parametri CloudWatch ed ethtool. Il `ethtool` comando è incluso nel driver ENA e permette di utilizzare parametri relativi alla rete aggiuntivi che possono essere pubblicate come parametri personalizzati in CloudWatch. Gli adattatori elastici di rete (ENA) Amazon offrono un'ulteriore ottimizzazione, migliorando il throughput per le tue istanze all'interno di un gruppo di posizionamento cluster. Elastic Fabric Adapter (EFA) è un'interfaccia di rete per le istanze Amazon EC2 che consente di eseguire carichi di lavoro che richiedono elevati livelli di comunicazioni internodi su vasta scala su AWS. Le istanze ottimizzate per Amazon EBS utilizzano uno stack di configurazione ottimizzato e forniscono un'ulteriore capacità dedicata per incrementare l'I/O di Amazon EBS.

Risorse

Documenti correlati:

- Application Load Balancer
- Reti avanzate EC2 su Linux
- Reti avanzate EC2 su Windows
- Gruppi di collocamento EC2
- Abilitazione delle reti avanzate con l'Adattatore elastico di rete (ENA) sulle istanze Linux
- Network Load Balancer
- Prodotti di rete con AWS
- Transitioning to Latency-Based Routing in Amazon Route 53
- Endpoint VPC
- Log di flusso VPC

Video correlati:

- AWS re:Invent 2023 – Ready for what's next? Designing networks for growth and flexibility
- AWS re:Invent 2023 – Advanced VPC designs and new capabilities
- AWS re:Invent 2023 – A developer's guide to cloud networking
- AWS re:Invent 2022 – Dive deep on AWS networking infrastructure
- AWS re:Invent 2019 – Connectivity to AWS and hybrid AWS network architectures
- AWS re:Invent 2018 – Optimizing Network Performance for Amazon EC2 Instances
- AWS Global Accelerator

Esempi correlati:

- AWS Transit Gateway and Scalable Security Solutions
- Workshop sulle reti AWS
- Observing and diagnosing your network
- Finding and addressing network misconfigurations on AWS

PERF04-BP03 Scegli la connettività dedicata o la VPN appropriata per il tuo carico di lavoro

Quando hai bisogno di una connettività ibrida per connettere risorse on-premises e cloud, assicurati di avere una larghezza di banda adeguata per soddisfare i tuoi requisiti di prestazione. Fai una

stima dei requisiti di larghezza di banda e latenza per il carico di lavoro ibrido. I valori calcolati determineranno le tue esigenze di dimensionamento.

Anti-pattern comuni:

- Valutazione delle soluzioni VPN solo per i tuoi requisiti di crittografia di rete.
- Non vengono valutate opzioni di backup o di connettività ridondante.
- Non è possibile identificare tutti i requisiti del carico di lavoro (esigenze di crittografia, protocollo, larghezza di banda e traffico).

Vantaggi dell'adozione di questa best practice: la selezione e la configurazione di soluzioni di connettività appropriate migliorano l'affidabilità del carico di lavoro e massimizzano le prestazioni. L'identificazione di requisiti del carico di lavoro, la pianificazione anticipata e la valutazione di soluzioni ibride ti permetteranno di ridurre al minimo le costose modifiche alla rete fisica e i costi operativi, migliorando al contempo il time-to-value.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Sviluppa un'architettura di rete ibrida basata sui tuoi requisiti di larghezza di banda. Direct Connect consente di connettere la rete on-premises in privato con AWS. È utile quando hai bisogno di larghezza di banda elevata, bassa latenza e di mantenere le prestazioni coerenti. Una connessione VPN permette di connettersi in modo sicuro su Internet. Viene utilizzata quando è necessaria solo una connessione temporanea, quando il costo è un fattore importante o come misura di contingenza in attesa che venga stabilita una connettività di rete fisica resiliente mentre Direct Connect è in uso.

Se i tuoi requisiti di larghezza di banda sono elevati, potresti prendere in considerazione l'utilizzo di più Direct Connect o di servizi di VPN. Il traffico può essere bilanciato in termini di carico tra i servizi, ma il bilanciamento del carico tra Direct Connect e VPN è sconsigliato a causa delle differenze di latenza e larghezza di banda.

Passaggi dell'implementazione

- Calcola i requisiti di larghezza di banda e latenza delle tue app esistenti.
  - Per i carichi di lavoro esistenti che vengono spostati in AWS, utilizza i dati raccolti dai sistemi di monitoraggio di rete interni.

    - Per i carichi di lavoro nuovi o esistenti per i quali non sono disponibili dati di monitoraggio, consulta i proprietari dei prodotti per definire metriche sulle prestazioni adeguate e offrire un'esperienza utente soddisfacente.
- Scegli una connessione dedicata o una VPN come opzione di connettività. A seconda di tutti i requisiti del carico di lavoro (esigenze di crittografia, larghezza di banda e traffico), puoi scegliere AWS Direct Connect o Site-to-Site VPN (o entrambi). Il diagramma seguente può aiutarti a scegliere il tipo di connessione appropriato.
    - AWS Direct Connect fornisce connettività dedicata all'ambiente AWS da 50 Mbps fino a 100 Gbps, utilizzando connessioni dedicate od ospitate. In questo modo, disporrai di latenza gestita e controllata, nonché di larghezza di banda assegnata, in modo che il carico di lavoro possa connettersi con efficienza ad altri ambienti. Ricorrendo a partner AWS Direct Connect, otterrai connettività end-to-end da più ambienti, per una rete estesa con prestazioni coerenti. AWS permette di dimensionare la larghezza di banda di connessione Direct Connect usando connettività nativa a 100 Gbps, gruppi di aggregazione di collegamenti (LAG, Link Aggregation Group) o instradamento ECMP (Equal-Cost Multipath) con BGP.
    - AWS VPN Site-to-Site offre un servizio VPN gestito che supporta il protocollo IPSec (Internet Protocol security). Quando viene creata una connessione VPN, ogni connessione include due tunnel per la disponibilità elevata.
- Consulta la documentazione AWS per scegliere l'opzione di connettività appropriata:
    - Se decidi di utilizzare Direct Connect, seleziona la larghezza di banda appropriata per la tua connettività.
    - In caso di utilizzo di una AWS Site-to-Site VPN tra più posizioni per connetterti a una Regione AWS, utilizza una connessione Site-to-Site VPN accelerata per migliorare le prestazioni di rete.
    - Se la progettazione della rete è costituita da una connessione VPN IPSec tramite AWS Direct Connect, prendi in considerazione l'utilizzo di VPN con indirizzo IP privato per migliorare la sicurezza e ottenere la segmentazione. AWS La VPN Site-to-Site Site con indirizzo IP privato viene implementata su un'interfaccia virtuale di transito (VIF).
    - AWS Direct Connect SiteLink consente di creare connessioni ridondanti e a bassa latenza tra i data center in tutto il mondo inviando dati lungo il percorso più veloce tra sedi AWS Direct Connect, bypassando Regioni AWS.
- Convalida la configurazione della connettività prima di eseguire l'implementazione in produzione. Esegui test di sicurezza e prestazioni per assicurarti di soddisfare i requisiti di larghezza di banda, affidabilità, latenza e conformità.
- Monitora regolarmente le prestazioni e l'utilizzo della connettività e ottimizzali, se necessario.

Diagramma di flusso per le prestazioni deterministiche

Risorse

Documenti correlati:

- Prodotti di rete con AWS
- AWS Transit Gateway
- Endpoint VPC
- Realizzazione di un'infrastruttura di reti multi-VPC sicura e scalabile
- Client VPN

Video correlati:

- AWS re:Invent 2023 – Building hybrid network connectivity with AWS
- AWS re:Invent 2023 – Secure remote connectivity to AWS
- AWS re:Invent 2022 – Optimizing performance with Amazon CloudFront
- AWS re:Invent 2019 – Connectivity to AWS and hybrid AWS network architectures
- AWS re:Invent 2020 – AWS Transit Gateway Connect

Esempi correlati:

- AWS Transit Gateway and Scalable Security Solutions
- AWS Workshop sulle reti

PERF04-BP04 Utilizzo del bilanciamento del carico per distribuire il traffico su più risorse

Distribuisci il traffico tra varie risorse o servizi affinché il carico di lavoro possa trarre vantaggio dall'elasticità fornita dal cloud. Puoi anche utilizzare il bilanciamento del carico per la terminazione dell'offloading della crittografia al fine di migliorare le prestazioni, l'affidabilità e gestire e instradare il traffico in modo efficiente.

Anti-pattern comuni:

- Scelta del tipo di sistema di bilanciatore del carico senza tenere conto dei requisiti del carico di lavoro.
- Mancato utilizzo delle funzionalità del bilanciatore del carico per l'ottimizzazione delle prestazioni.
- Esposizione diretta del carico di lavoro a Internet senza un bilanciatore del carico.
- Instradi tutto il traffico Internet attraverso i bilanciatori del carico esistenti.
- Utilizzi il bilanciamento del carico TCP generico e fai in modo che ogni nodo di calcolo gestisca la crittografia SSL.

Vantaggi dell'adozione di questa best practice: un bilanciatore del carico gestisce il carico variabile del traffico dell'applicazione in una o più zone di disponibilità e consente alta disponibilità, dimensionamento automatico e un migliore utilizzo del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I bilanciatori del carico operano come punto di ingresso per il carico di lavoro, dal quale distribuiscono il traffico alle destinazioni di backend, come istanze di calcolo o container per migliorarne l'utilizzo.

La scelta del tipo corretto di bilanciatore del carico è il primo passaggio per ottimizzare l'architettura. Per iniziare, elenca le caratteristiche del carico di lavoro, tra cui protocollo (TCP, HTTP, TLS o WebSocket), tipo di destinazione (istanze, container o servizi serverless), requisiti dell'applicazione (connessioni a esecuzione prolungata, autenticazione utente o persistenza) e ubicazione (regione, zona locale, Outpost o isolamento zonale).

AWS fornisce diversi modelli di bilanciamento del carico per le tue applicazioni. Application Load Balancer è l'ideale per il bilanciamento del carico del traffico HTTP e HTTPS. Inoltre, offre l'instradamento avanzato delle richieste, dedicato alla distribuzione delle architetture applicative moderne, fra cui microservizi e container.

Network Load Balancer è l'ideale per il bilanciamento del carico del traffico TCP, in cui sono richieste prestazioni elevatissime. È in grado di gestire milioni di richieste al secondo, mantenendo al contempo latenze ridottissime. Inoltre, è ottimizzato per la gestione degli schemi di traffico improvvisi e incostanti.

Elastic Load Balancing fornisce la gestione integrata dei certificati e la decrittografia SSL/TLS, offrendoti la flessibilità di gestire centralmente le impostazioni SSL del bilanciatore del carico e di sollevare il carico di lavoro dall'utilizzo intensivo della CPU.

Dopo aver scelto il bilanciatore del carico appropriato, puoi iniziare a utilizzarne le funzionalità per ridurre la quantità di attività che deve svolgere il backend per distribuire il traffico.

Ad esempio, usando Application Load Balancer (ALB) e Network Load Balancer (NLB), puoi eseguire l'offload della crittografia SSL/TLS, il che costituisce un'opportunità per evitare il completamento dell'handshake TLS a elevato utilizzo di CPU da parte delle destinazioni e migliorare anche la gestione dei certificati.

Se configurato nel bilanciatore del carico, l'offload SSL/TLS diventa responsabile della crittografia del traffico da e verso i client, distribuendo il traffico non crittografato ai backend, liberando le risorse backend e migliorando il tempo di risposta per i client.

Application Load Balancer può anche distribuire traffico HTTP/2 senza che questo debba essere supportato nelle destinazioni. Questa semplice decisione può migliorare il tempo di risposta dell'applicazione, in quanto HTTP/2 usa connessioni TCP in modo più efficiente.

Nel definire l'architettura, è bene tenere conto dei requisiti di latenza del carico di lavoro. Ad esempio, se un'applicazione è sensibile alla latenza, è possibile scegliere di usare Network Load Balancer, che offre latenze estremamente ridotte. In alternativa, è possibile decidere di avvicinare il carico di lavoro ai clienti sfruttando Application Load Balancer nelle zone locali AWS o addirittura AWS Outposts.

Un altro aspetto di cui tenere conto per i carichi di lavoro sensibili alla latenza è il bilanciamento del carico tra zone. Con il bilanciamento del carico tra zone, ogni nodo del bilanciatore del carico distribuisce il traffico tra le destinazioni registrate in tutte le zone di disponibilità autorizzate.

Usa Auto Scaling integrato con il bilanciatore del carico. Uno degli aspetti principali di un sistema con prestazioni efficienti riguarda il dimensionamento corretto delle risorse backend. A questo scopo, puoi utilizzare integrazioni dei bilanciatori del carico per le risorse di destinazione backend. Usando l'integrazione dei bilanciatori del carico con gruppi Auto Scaling, le destinazioni vengono aggiunte o rimosse nel e dal bilanciatore del carico in base alle esigenze, in risposta al traffico in ingresso. I bilanciatori del carico possono integrarsi anche con Amazon ECS e Amazon EKS per carichi di lavoro distribuiti in container.

- Amazon ECS: bilanciamento del carico di servizio
- Bilanciamento del carico di applicazione su Amazon EKS
- Bilanciamento del carico di rete su Amazon EKS

Passaggi dell'implementazione

- Definisci i tuoi requisiti di bilanciamento del carico, tra cui volume di traffico, disponibilità e scalabilità delle applicazioni.
- Scegli il tipo di sistema di bilanciatore del carico giusto per la tua applicazione.
  - Utilizza Application Load Balancer per i carichi di lavoro HTTP/HTTPS.
  - Utilizza Network Load Balancer per carichi di lavoro non HTTP in esecuzione su TCP o UDP.
  - Usa una combinazione dei due sistemi (ALB come destinazione di NLB) per sfruttare le funzionalità di entrambi i prodotti. Ad esempio, puoi scegliere questa opzione se vuoi usare gli indirizzi IP statici dell'NLB insieme all'instradamento basato su intestazione HTTP dell'ALB, oppure se vuoi esporre il carico di lavoro HTTP a un AWS PrivateLink.
  - Per un confronto completo dei bilanciatori del carico, consulta la tabella di confronto dei prodotti ELB.
- Se possibile, utilizza l'offload SSL/TLS.
  - Configura gli ascoltatori HTTPS/TLS con Application Load Balancer e Network Load Balancer integrati con AWS Certificate Manager.
  - Alcuni carichi di lavoro possono richiedere la crittografia end-to-end per motivi di conformità. In questo caso, è necessario consentire la crittografia nelle destinazioni.
  - Per le best practice in materia di sicurezza, consulta SEC09-BP02 Applicazione della crittografia dei dati in transito.
- Seleziona l'algoritmo di instradamento corretto (solo ALB)
  - L'algoritmo di instradamento può fare la differenza per quanto riguarda l'uso corretto delle destinazioni backend e, di conseguenza, l'impatto sulle prestazioni. Ad esempio, ALB offre due opzioni per gli algoritmi di instradamento:
  - Numero minimo di richieste in sospeso: usa questa opzione per ottenere una migliore distribuzione del carico nelle destinazioni backend nei casi in cui le richieste per l'applicazione variano per complessità o le destinazioni variano per capacità di elaborazione.
  - Round robin: usa questa opzione quando le richieste e le destinazioni sono simili o se devi distribuire equamente le richieste tra le destinazioni.
- Valuta se usare l'isolamento tra zone o quello zonale.
  - Disattiva l'isolamento tra zone (usando l'isolamento zonale) per migliorare la latenza e in caso di domini con errori di zona. La funzione è disattivata per impostazione predefinita in NLB e in ALB è possibile disattivarla per gruppo di destinazione.

    - Attiva l'isolamento tra zone per ottenere disponibilità e flessibilità maggiori. L'isolamento tra zone è disattivato per impostazione predefinita in ALB e in NLB è possibile attivarlo per gruppo di destinazione.
- Attiva keep-alive HTTP per i carichi di lavoro HTTP (solo ALB). Con questa funzionalità, il bilanciatore del carico può riutilizzare le connessioni backend fino allo scadere del timeout del keep-alive, migliorando la richiesta HTTP e il tempo di risposta e riducendo anche l'utilizzo delle risorse nelle destinazioni backend. Per informazioni sulla configurazione per Apache e Nginx, consulta Quali sono le impostazioni ottimali per utilizzare Apache o NGINX come server di backend per ELB?
- Attiva il monitoraggio del tuo bilanciatore del carico.
    - Attiva i log di accesso per Application Load Balancer e Network Load Balancer.
    - I campi principali da considerare per l'ALB sono `request_processing_time`, `request_processing_time` e `response_processing_time`.
    - I campi principali da considerare per l'NLB sono `connection_time` e `tls_handshake_time`.
    - Preparati a eseguire query sui log quando necessario. Puoi usare Amazon Athena per eseguire query sui log ALB e sui log NLB.
    - Crea allarmi per metriche correlate alle prestazioni, come `TargetResponseTime` per ALB.

Risorse

Documenti correlati:

- Tabella di confronto dei prodotti ELB
- Infrastruttura globale di AWS
- Improving Performance and Reducing Cost Using Availability Zone Affinity
- Step by step for Log Analysis with Amazon Athena
- Querying Application Load Balancer logs
- Monitor your Application Load Balancers
- Monitor your Network Load Balancer
- Use Elastic Load Balancing to distribute traffic across the instances in your Auto Scaling group

Video correlati:

- AWS re:Invent 2023: What can networking do for your application?
- AWS re:Inforce 20: How to use Elastic Load Balancing to enhance your security posture at scale
- AWS re:Invent 2018: Elastic Load Balancing: Deep Dive and Best Practices
- AWS re:Invent 2021 - How to choose the right load balancer for your AWS workloads
- AWS re:Invent 2019: Get the most from Elastic Load Balancing for different workloads

Esempi correlati:

- Gateway Load Balancer
- CDK and CloudFormation samples for Log Analysis with Amazon Athena

PERF04-BP05 Scelta dei protocolli di rete per migliorare le prestazioni

Prendi decisioni sui protocolli per la comunicazione tra sistemi e reti in base all'impatto sulle prestazioni del carico di lavoro.

Esiste una relazione tra latenza e larghezza di banda per ottenere il throughput desiderato. Se per il trasferimento file si usa il protocollo TCP, latenze più elevate molto probabilmente ridurranno il throughput complessivo. Alcuni approcci risolvono questo problema tramite l'ottimizzazione del TCP e l'utilizzo di protocolli di trasferimento ottimizzati, ma una soluzione prevede l'utilizzo del protocollo User Datagram Protocol (UDP).

Anti-pattern comuni:

- Puoi utilizzare il TCP per tutti i carichi di lavoro, indipendentemente dai requisiti prestazionali.

Vantaggi dell'adozione di questa best practice: la verifica del protocollo adeguato per la comunicazione tra utenti e componenti del carico di lavoro contribuisce a migliorare l'esperienza utente complessiva per le applicazioni. Ad esempio, l'UDP senza connessione garantisce velocità elevata, ma non offre ritrasmissione o elevata affidabilità. Il TCP è un protocollo completo, ma richiede un sovraccarico maggiore per l'elaborazione dei pacchetti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Se hai la possibilità di scegliere protocolli diversi per la tua applicazione e hai esperienza in questo campo, ottimizza l'applicazione e l'esperienza dell'utente finale utilizzando un protocollo diverso.

Tieni conto che questo approccio presenta notevoli difficoltà e dovrebbe essere tentato solo dopo l'ottimizzazione dell'applicazione in altri modi.

Un aspetto fondamentale per il miglioramento delle prestazioni del tuo carico di lavoro consiste nell'identificare i requisiti in termini di latenza e throughput, quindi scegliere i protocolli di rete che ottimizzano le prestazioni.

Quando valutare se usare TCP

Il protocollo TCP permette la trasmissione affidabile dei dati e può essere usato per la comunicazione tra i componenti del carico di lavoro quando l'affidabilità e la garanzia di trasmissione dei dati sono due aspetti importanti. Molte applicazioni Web usano protocolli basati su TCP, come HTTP e HTTPS, per aprire socket TCP per la comunicazione tra i componenti dell'applicazione. Il TCP viene comunemente usato per il trasferimento di dati di posta elettronica e di file, in quanto è un meccanismo di trasferimento semplice e affidabile tra i componenti dell'applicazione. L'uso di TLS con TCP può aggiungere un certo sovraccarico alla comunicazione, il che produce maggiore latenza e throughput inferiore, ma presenta come vantaggio una maggiore sicurezza. Il sovraccarico è dovuto perlopiù al processo di handshake, il cui completamento può richiedere diversi round trip. Al termine del processo di handshake, il sovraccarico dovuto alla crittografia e alla decrittografia dei dati è relativamente ridotto.

Quando valutare se usare UDP

UDP è un protocollo di tipo connection-less (senza connessione) e di conseguenza è ideale per applicazioni che necessitano di una trasmissione veloce ed efficiente, ad esempio per i log, il monitoraggio e i dati VoIP. Valuta se usare UDP anche se in presenza di componenti del carico di lavoro che rispondono a piccole query provenienti da grandi quantità di client per garantire prestazioni ottimali del carico di lavoro. Datagram Transport Layer Security (DTLS) è l'equivalente UDP di Transport Layer Security (TLS). In caso di utilizzo di DTLS con UDP, il sovraccarico è dovuto alla crittografia e alla decrittografia dei dati, in quanto il processo di handshake è semplificato. DTLS aggiunge anche un piccolo sovraccarico ai pacchetti UDP, poiché comprende altri campi per indicare i parametri di sicurezza e rilevare la manomissione.

Quando valutare se usare SRD

SRD (Scalable Reliable Datagram) è un protocollo di trasporto di rete ottimizzato per carichi di lavoro a elevato throughput grazie alla sua capacità di bilanciatore del carico del traffico tra più percorsi e di recuperare rapidamente dalla perdita di pacchetti e da errori di collegamento. Di conseguenza, SRD è ideale per carichi di lavoro di calcolo ad alte prestazioni (HPC) che richiedono

comunicazioni tra nodi di calcolo a throughput elevato e a bassa latenza. Possono essere incluse attività di elaborazione in parallelo come la simulazione, la modellazione e l'analisi dei dati che implicano il trasferimento di grandi quantità di dati tra nodi.

Passaggi dell'implementazione

- Utilizzare i servizi AWS Global Accelerator e AWS Transfer Family per migliorare il throughput delle applicazioni di trasferimento file online. Il servizio AWS Global Accelerator ti permette di ottenere latenze inferiori tra i dispositivi client e il carico di lavoro in AWS. Con AWS Transfer Family puoi usare protocolli basati su TCP come SFTP (Secure Shell File Transfer Protocol) e FTPS (File Transfer Protocol over SSL) per scalare e gestire i trasferimenti file in servizi di archiviazione AWS in tutta sicurezza.
- Usa la latenza di rete per determinare se TCP sia il protocollo appropriato per la comunicazione tra componenti del carico di lavoro. Se la latenza di rete tra l'applicazione client e il server è elevata, il processo di handshake a tre vie tramite TCP può richiedere tempo, influendo sulla velocità di risposta dell'applicazione. Per misurare la latenza di rete, puoi usare, ad esempio, le metriche tempo di acquisizione al primo byte (TTFB) e tempo di andata e ritorno (RTT) Se il tuo carico di lavoro offre contenuti dinamici agli utenti, prendi in considerazione l'utilizzo di Amazon CloudFront, che stabilisce una connessione persistente a ciascuna origine per il contenuto dinamico in modo da eliminare il tempo di configurazione della connessione, che altrimenti rallenterebbe ogni richiesta client.
- L'uso di TLS con TCP o UDP può causare maggiore latenza e minore throughput per il carico di lavoro a causa dell'impatto della crittografia e della decrittografia. Per tali carichi di lavoro, prendi in considerazione l'offload SSL/TLS su Elastic Load Balancing per migliorare le prestazioni del carico di lavoro permettendo al bilanciatore del carico di gestire la crittografia e la decrittografia SSL/TLS invece di predisporre a questo scopo istanze backend. In questo modo, puoi ridurre l'utilizzo della CPU sulle istanze backend, migliorando le prestazioni e aumentando la capacità.
- Usa Network Load Balancer (NLB) per implementare servizi basati sul protocollo UDP, tra cui autenticazione e autorizzazione, log, DNS, IoT e streaming di contenuti multimediali, in modo da migliorare prestazioni e affidabilità del carico di lavoro. L'NLB distribuisce il traffico UDP in ingresso tra più destinazioni, permettendo di scalare orizzontalmente il carico di lavoro, incrementare la capacità e diminuire il sovraccarico su un'unica destinazione.
- Per i carichi di lavoro di calcolo ad alte prestazioni (HPC), prendi in considerazione l'utilizzo della funzionalità Adattatore elastico di rete (ENA) Express che sfrutta il protocollo SRD per migliorare le prestazioni di rete fornendo una maggiore larghezza di banda a flusso singolo (25 Gbps) e una latenza di coda inferiore (99,9 percentile) per il traffico di rete tra istanze EC2.

- Usa Application Load Balancer (ALB) per instradare e bilanciare il traffico gRPC (Remote Procedure Call) tra componenti del carico di lavoro o tra client e servizi gRPC e per bilanciarne il carico. gRPC usa il protocollo HTTP/2 basato su TCP per il trasporto e fornisce vantaggi in termini di prestazioni, tra cui un impatto di rete minore, la compressione, la serializzazione binaria efficiente, il supporto per diversi linguaggi e lo streaming bidirezionale.

Risorse

Documenti correlati:

- How to route UDP traffic into Kubernetes
- Application Load Balancer
- Reti avanzate EC2 su Linux
- Reti avanzate EC2 su Windows
- Gruppi di collocamento EC2
- Abilitazione delle reti avanzate con l'Adattatore elastico di rete (ENA) sulle istanze Linux
- Network Load Balancer
- Prodotti di rete con AWS
- Passaggio all'instradamento basato sulla latenza in Amazon Route 53
- Endpoint VPC

Video correlati:

- AWS re:Invent 2022 – Scaling network performance on next-gen Amazon Elastic Compute Cloud instances
- AWS re:Invent 2022 – Application networking foundations

Esempi correlati:

- AWS Transit Gateway and Scalable Security Solutions
- Workshop sulle reti AWS

PERF04-BP06 Scegli la posizione del carico di lavoro in base ai requisiti di rete

Valuta le opzioni per il posizionamento delle risorse in modo da diminuire la latenza di rete e migliorare il throughput, fornendo un'esperienza utente ottimale attraverso la riduzione dei tempi di caricamento delle pagine e di trasferimento dei dati.

Anti-pattern comuni:

- Consolidamento di tutte le risorse del carico di lavoro in un'unica posizione geografica.
- Scelta della regione più vicina alla propria posizione, ma non al carico di lavoro dell'utente finale.

Vantaggi dell'adozione di questa best practice: l'esperienza utente è fortemente condizionata dalla latenza tra utente e applicazione. Utilizzando una rete globale appropriata Regioni AWS e AWS privata, è possibile ridurre la latenza e offrire un'esperienza migliore agli utenti remoti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Le risorse, come le EC2 istanze Amazon, vengono collocate nelle Availability Zones within Regioni AWS, AWS Local Zones o AWS Wavelengthnelle zone. AWS Outposts La scelta della posizione influisce su latenza di rete e throughput dall'ubicazione di un utente specifico. I servizi edge come Amazon CloudFront AWS Global Acceleratorpossono essere utilizzati anche per migliorare le prestazioni di rete memorizzando nella cache i contenuti nelle edge location o fornendo agli utenti un percorso ottimale per il carico di lavoro attraverso la rete AWS globale.

Amazon EC2 fornisce gruppi di collocamento per il networking. Un gruppo di collocazione è un raggruppamento logico di istanze per ridurre la latenza. L'utilizzo di gruppi di collocamento con tipi di istanze supportati e un Elastic Network Adapter (ENA) consente ai carichi di lavoro di partecipare a una rete a 25 Gbps a bassa latenza e con jitter ridotto. I gruppi di collocazione sono consigliati per i carichi di lavoro che traggono beneficio da reti a bassa latenza, throughput di rete elevato o entrambi.

I servizi sensibili alla latenza vengono forniti nelle sedi periferiche utilizzando una rete AWS globale, come Amazon. CloudFront Queste edge location forniscono in genere servizi come Content Delivery Network (CDN) e Domain Name System (). DNS Disponendo di questi servizi all'edge, i carichi di lavoro possono rispondere con bassa latenza alle richieste di contenuto o DNS risoluzione. Inoltre, possono offrire servizi geografici come la geotargetizzazione dei contenuti (ossia fornire contenuti diversi in base alla posizione dell'utente finale) o l'instradamento basato sulla latenza, per indirizzare gli utenti alla regione più vicina (latenza minima).

Usa i servizi edge per ridurre la latenza e abilitare la memorizzazione nella cache dei contenuti. Configura correttamente il controllo della cache per entrambi DNS eHTTP/HTTPSper ottenere il massimo vantaggio da questi approcci.

Passaggi dell'implementazione

- Acquisisci informazioni sul traffico IP in entrata e in uscita dalle interfacce di rete.
  - Registrazione del traffico IP utilizzando VPC Flow Logs
  - Come viene preservato l'indirizzo IP del client in AWS Global Accelerator
- Analizza i modelli di accesso alla rete nel tuo carico di lavoro per capire come gli utenti usano la tua applicazione.
  - Utilizza strumenti di monitoraggio, come Amazon CloudWatch e AWS CloudTrail, per raccogliere dati sulle attività di rete.
  - Analizza i dati per identificare il modello di accesso alla rete.
- Seleziona regioni appropriate per l'implementazione del carico di lavoro in base ai seguenti elementi chiave:
  - Ubicazione dei dati per le applicazioni a uso intensivo di dati, ad esempio applicazioni di big data e machine learning, il codice dell'applicazione dovrebbe essere eseguito il più vicino possibile ai dati.
  - Ubicazione degli utenti: per le applicazioni rivolte agli utenti, scegli una regione o più regioni vicine agli utenti del carico di lavoro.
  - Altri vincoli: prendi in considerazione vincoli come costi e conformità, come illustrato in What to Consider when Selecting a Region for your Workloads.
- Usa le zone locali AWS per eseguire carichi di lavoro come il rendering video. Le zone locali consentono di sfruttare i vantaggi derivanti dalla disponibilità di risorse di calcolo e archiviazione più vicine agli utenti finali.
- Usa AWS Outposts per carichi di lavoro che devono rimanere on-premises, ma vuoi che vengano eseguiti in modo ottimale con il resto degli altri carichi di lavoro in AWS.
- Applicazioni come lo streaming video in diretta ad alta risoluzione, l'audio ad alta fedeltà e la realtà aumentata o la realtà virtuale (AR/VR) richiedono dispositivi 5G. ultra-low-latency Per tali applicazioni, considera. AWS Wavelength AWS Wavelength incorpora servizi di AWS elaborazione e archiviazione nelle reti 5G, fornendo un'infrastruttura di edge computing mobile per lo sviluppo, l'implementazione e la scalabilità delle applicazioni. ultra-low-latency
- Usa la cache locale o le soluzioni di caching AWS per i dati di frequente utilizzo per migliorare le performance, ridurre lo spostamento dei dati e minimizzare l'impatto ambientale.

Servizio	Quando usare
Amazon CloudFront	Utilizzalo per memorizzare nella cache contenuti statici come immagini, script e video, nonché contenuti dinamici come API risposte o applicazioni web.
Amazon ElastiCache	Usalo per memorizzare nella cache i contenuti per le applicazioni Web.
DynamoDB Accelerator	Usalo per aggiungere accelerazione in memoria alle tabelle DynamoDB.

- Utilizza servizi in grado di supportarti nell'esecuzione del codice in posizioni più vicine agli utenti del carico di lavoro, come i seguenti:

Servizio	Quando usare
Lambda@Edge	Usalo per operazioni a uso intensivo di risorse di calcolo eseguite quando gli oggetti non si trovano nella cache.
CloudFront Funzioni Amazon	Utilizzalo per casi d'uso semplici come richieste HTTP (s) o manipolazioni di risposte che possono essere avviate da funzioni di breve durata.
AWS IoT Greengrass	Usale per eseguire la memorizzazione nella cache di risorse di calcolo, messaggistica e dati per i dispositivi connessi.

- Alcune applicazioni richiedono punti di ingresso fissi o prestazioni più elevate attraverso la riduzione della latenza di ricezione del primo byte e l'instabilità e l'aumento del throughput. Queste applicazioni possono trarre vantaggio da servizi di rete che forniscono indirizzi IP anycast statici e TCP terminazioni in postazioni periferiche. AWS Global Acceleratorpossono migliorare le prestazioni delle applicazioni fino al 60% e fornire un failover rapido per architetture multiregionali. AWS Global Accelerator fornisce indirizzi IP anycast statici che fungono da punto di ingresso fisso

per le applicazioni ospitate in una o più applicazioni. Regioni AWS Questi indirizzi IP consentono al traffico di entrare nella rete AWS globale il più vicino possibile agli utenti. AWS Global Accelerator riduce il tempo di configurazione iniziale della connessione stabilendo una TCP connessione tra il client e la AWS edge location più vicina al client. Rivedi l'utilizzo di AWS Global Accelerator per migliorare le prestazioni dei tuoi UDP carichi di lavoroTCP/e fornire un failover rapido per architetture multiregionali.

Risorse

Best practice correlate:

- COST07-BP02 Implementazione delle regioni in base ai costi
- COST08-BP03 Implementazione di servizi per ridurre i costi di trasferimento dei dati
- REL10-BP01 Implementa il carico di lavoro in più sedi
- REL10-BP02 Seleziona le posizioni appropriate per l'implementazione in più sedi
- SUS01-BP01 Scegli la regione in base ai requisiti aziendali e agli obiettivi di sostenibilità
- SUS02-BP04 Ottimizza il posizionamento geografico dei carichi di lavoro in base ai requisiti di rete
- SUS04-BP07 Riduci al minimo lo spostamento dei dati tra le reti

Documenti correlati:

- AWS Infrastruttura globale
- AWS Local Zones e AWS Outposts scelta della tecnologia giusta per il tuo carico di lavoro edge
- Placement groups
- AWS Local Zones
- AWS Outposts
- AWS Wavelength
- Amazon CloudFront
- AWS Global Accelerator
- AWS Direct Connect
- AWS Site-to-Site VPN
- Amazon Route 53

Video correlati:

- AWS Video esplicativo su Local Zones
- AWS Outposts: Overview and How it Works
- AWS re:Invent 2023 - Una strategia di migrazione per carichi di lavoro edge e locali
- AWS re:Invent 2021 -: Portare l'esperienza in sede AWS OutpostsAWS
- AWS re:Invent 2020:: Esegui app con latenza AWS Wavelength ultra bassa su 5G Edge
- AWS re:Invent 2022 - AWS Local Zones: creazione di applicazioni per un edge distribuito
- AWS re:Invent 2021 - Creazione di siti Web a bassa latenza con Amazon CloudFront
- AWS re:Invent 2022 - Migliora le prestazioni e la disponibilità con AWS Global Accelerator
- AWS re:Invent 2022 - Costruisci la tua rete WAN utilizzando AWS
- AWS re:Invent 2020: gestione globale del traffico con Amazon Route 53

Esempi correlati:

- AWS Global Accelerator Workshop sul routing personalizzato
- Handling Rewrites and Redirects using Edge Functions

PERF04-BP07 Ottimizzazione della configurazione di rete in base alle metriche

Usa i dati raccolti e analizzati per prendere decisioni informate riguardo l'ottimizzazione della configurazione della tua rete.

Anti-pattern comuni:

- Si ritiene che tutti i problemi relativi alle prestazioni siano correlati all'applicazione.
- Verifica delle prestazioni di rete solo da una posizione vicina a quella in cui è stato distribuito il carico di lavoro.
- Uso di configurazioni predefinite per tutti i servizi di rete.
- Provisioning in eccesso di risorse di rete per fornire capacità sufficiente.

Vantaggi dell'adozione di questa best practice: la raccolta delle metriche necessarie per la rete AWS e l'implementazione di strumenti di monitoraggio di rete permettono di identificare le prestazioni di rete e ottimizzare le configurazioni di rete.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Il monitoraggio del traffico da e verso VPC, sottoreti o interfacce di rete è essenziale per identificare come utilizzare risorse di rete AWS e ottimizzare le configurazioni di rete. Usando i seguenti strumenti di rete AWS, puoi esaminare ulteriormente le informazioni sull'utilizzo del traffico, sull'accesso alla rete e sui log.

Passaggi dell'implementazione

- Identifica le metriche delle prestazioni fondamentali da raccogliere, come la latenza o la perdita di pacchetti. AWS fornisce diversi strumenti che possono aiutarti a raccogliere queste metriche. Usando i seguenti strumenti, puoi esaminare ulteriormente le informazioni sull'utilizzo del traffico, sull'accesso alla rete e sui log:

Strumento AWS	Dove usarlo
Amazon VPC IP Address Manager.	Utilizza IPAM per pianificare, seguire e monitorare gli indirizzi IP per i carichi di lavoro AWS e on-premises. Si tratta di una best practice per ottimizzare l'utilizzo e l'allocazione degli indirizzi IP.
Log di flusso VPC	Usa log di flusso VPC per acquisire informazioni dettagliate sul traffico da e verso le interfacce di rete nei VPC. Con i log di flusso VPC, puoi diagnosticare regole dei gruppi di sicurezza eccessivamente restrittive o permissive e determinare la direzione del traffico da e verso le interfacce di rete.
Log di flusso AWS Transit Gateway	Utilizza i log di flusso AWS Transit Gateway per acquisire informazioni sul traffico IP in entrata e in uscita dai gateway di transito.
Log di query DNS	Crea log di informazioni sulle query DNS pubbliche o private ricevute da Route 53. Con i log DNS puoi ottimizzare le configurazioni

Strumento AWS	Dove usarlo
	DNS identificando il dominio e il sottodomi nio richiesto o le posizioni edge Route 53 che hanno risposto a query DNS.
Reachability Analyzer	Con Reachability Analyzer puoi analizzare la raggiungibilità della rete ed eseguirne il debug. Reachability Analyzer è uno strumento di analisi della configurazione che consente di eseguire test di connettività tra una risorsa di origine e una risorsa di destinazione nei VPC. Lo strumento in questione consente di verificare la corrispondenza fra configurazione e connettività desiderata.
Strumento di analisi degli accessi alla rete	È possibile utilizzare lo Strumento di analisi degli accessi alla rete per comprendere l'accesso di rete alle risorse. Puoi usare lo Strumento di analisi degli accessi alla rete per specificare i requisiti di accesso alla rete e identificare i potenziali percorsi di rete che non li soddisfano. Ottimizzando la configura zione di rete corrispondente, puoi determinare e verificare lo stato della rete e indicare se la rete su AWS soddisfa i requisiti di conformità.

Strumento AWS	Dove usarlo
Amazon CloudWatch	Utilizza Amazon CloudWatch e attiva le metriche opportune per le opzioni di rete. Assicurati di scegliere le metriche di rete corrette per il carico di lavoro. Ad esempio, puoi attivare le metriche per l'utilizzo degli indirizzi di rete del VPC, il gateway NAT del VPC, AWS Transit Gateway, il tunnel VPN, AWS Network Firewall, Elastic Load Balancing , e AWS Direct Connect. Il monitoraggio continuo delle metriche è una procedura utile per osservare e identificare lo stato e l'utilizzo della rete che semplifica l'ottimizzazione della configurazione di rete in base alle osservazi oni.
AWS Network Manager	Grazie a AWS Network Manager, puoi monitorare le prestazioni in tempo reale e cronologiche della rete globale AWS per scopi operativi e di pianificazione. Network Manager fornisce una latenza di rete aggregata tra Regioni AWS e zone di disponibilità e all'inter no di ciascuna zona di disponibilità, permetten doti di comprendere meglio la relazione fra prestazioni delle applicazioni e prestazioni della rete AWS sottostante.
Amazon CloudWatch RUM	Usa Amazon CloudWatch RUM per raccoglie re le metriche che ti consentono di ottenere approfondimenti utili per identificare, comprendere e migliorare l'esperienza utente.

- Identifica i top talker e gli schemi di traffico delle applicazioni utilizzando VPC e i log di flusso di AWS Transit Gateway.

- Valuta e ottimizza la tua attuale architettura di rete, inclusi VPC, sottoreti e routing. Ad esempio, puoi valutare come i diversi VPC per il peering o AWS Transit Gateway possono aiutarti a migliorare la rete nella tua architettura.
- Valuta i percorsi di instradamento nella tua rete per verificare che venga sempre utilizzato il percorso più breve tra le destinazioni. Lo Strumento di analisi degli accessi alla rete è utile in questa operazione.

Risorse

Documenti correlati:

- Public DNS query logging
- What is IPAM?
- What is Reachability Analyzer?
- What is Network Access Analyzer?
- CloudWatch metrics for your VPCs
- Optimize performance and reduce costs for network analytics with VPC Flow Logs in Apache Parquet format
- Monitoring your global and core networks with Amazon CloudWatch metrics
- Continuously monitor network traffic and resources

Video correlati:

- AWS re:Invent 2023 – A developer's guide to cloud networking
- AWS re:Invent 2023 – Ready for what's next? Designing networks for growth and flexibility
- AWS re:Invent 2023 – Advanced VPC designs and new capabilities
- AWS re:Invent 2022 – Dive deep on AWS networking infrastructure
- AWS re:Invent 2020 – Networking best practices and tips with the AWS Well-Architected Framework
- AWS re:Invent 2020 – Monitoring and troubleshooting network traffic

Esempi correlati:

- Workshop sulle reti AWS

- AWS Network Monitoring
- Observing and diagnosing your network on AWS
- Finding and addressing network misconfigurations on AWS

Processo e cultura

Questions

- PERF 5. In che modo le pratiche e la cultura dell'organizzazione contribuiscono all'efficienza delle prestazioni nel carico di lavoro?

PERF 5. In che modo le pratiche e la cultura dell'organizzazione contribuiscono all'efficienza delle prestazioni nel carico di lavoro?

Durante la fase di progettazione dei carichi di lavoro, esistono principi e pratiche che è possibile adottare per gestire al meglio carichi di lavoro cloud efficienti e ad alte prestazioni. Per adottare una cultura che promuova l'efficienza delle prestazioni dei carichi di lavoro cloud, prendi in considerazione questi principi e pratiche fondamentali:

Best practice

- PERF05-BP01 Individuazione degli indicatori chiave di prestazioni (KPI) per misurare l'integrità e le prestazioni del carico di lavoro
- PERF05-BP02 Uso di soluzioni di monitoraggio per comprendere le aree in cui le prestazioni sono più critiche
- PERF05-BP03 Definizione di un processo per migliorare le prestazioni del carico di lavoro
- PERF05-BP04 Load Esegui un test del tuo carico di lavoro
- PERF05-BP05 Uso dell'automazione per risolvere in modo proattivo i problemi relativi alle prestazioni
- PERF05-BP06 Conserva il carico di lavoro e i servizi up-to-date
- PERF05-BP07 Analisi dei parametri a intervalli regolari

PERF05-BP01 Individuazione degli indicatori chiave di prestazioni (KPI) per misurare l'integrità e le prestazioni del carico di lavoro

Individua gli indicatori chiave di prestazione (KPI) per misurare le prestazioni del carico di lavoro. I KPI consentono di misurare integrità e prestazioni di un carico di lavoro correlato a un obiettivo aziendale.

Anti-pattern comuni:

- Monitoraggio dei parametri a livello di sistema solo per avere una visione del carico di lavoro e mancata valutazione degli impatti aziendali di tali parametri.
- Si suppone che i KPI siano già in fase di pubblicazione e condivisi come dati parametrici standard.
- Mancata definizione di un KPI quantitativo e misurabile.
- Mancato allineamento dei KPI a obiettivi o strategie aziendali.

Vantaggi dell'adozione di questa best practice: l'individuazione di KPI specifici che rappresentino integrità e prestazioni del carico di lavoro aiuta ad allineare i team alle priorità e a definire risultati aziendali ottimali. La condivisione di tali metriche con tutti i reparti fornisce visibilità e allineamento su soglie, aspettative e impatto aziendale.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Gli indicatori chiave di prestazione consentono ai team aziendali e di ingegneri di allinearsi in termini di misurazione degli obiettivi e delle strategie e sul modo in cui questi fattori si combinano per produrre risultati aziendali. Ad esempio, il carico di lavoro di un sito Web può utilizzare il tempo di caricamento della pagina come indicazione delle prestazioni complessive. Questa metrica sarebbe uno dei vari punti dati che misurano l'esperienza dell'utente. Oltre a identificare le soglie di tempo di caricamento della pagina, occorre documentare il risultato atteso o il rischio aziendale in caso di mancato raggiungimento delle prestazioni ideali. Un lungo tempo di caricamento della pagina si ripercuote direttamente sugli utenti finali, peggiora la loro esperienza d'uso e può portare a una perdita di clienti. Quando definisci le soglie degli indicatori chiave di prestazione, devi combinare benchmark di settore e aspettative degli utenti finali. Ad esempio, se l'attuale benchmark del settore prevede il caricamento di una pagina Web entro un periodo di tempo di due secondi, ma gli utenti finali si aspettano che la pagina Web venga caricata entro un periodo di tempo di un secondo, allora devi prendere in considerazione entrambi i dati al momento di stabilire l'indicatore chiave di prestazione (KPI).

Il team deve valutare i KPI del carico di lavoro, utilizzando dati granulari in tempo reale e dati cronologici di riferimento, e creare pannelli di controllo che eseguano calcoli metrici sui dati KPI per ricavare informazioni operative e di utilizzo. I KPI devono essere documentati e includere le soglie che supportano gli obiettivi e le strategie aziendali, mappati sui parametri da monitorare. Gli indicatori chiave di prestazione devono essere riesaminati in caso di cambiamento di obiettivi aziendali, strategie o requisiti degli utenti finali.

Passaggi dell'implementazione

- Identifica le parti interessate: identifica e documenta le principali parti interessate aziendali, compresi i team di sviluppo e operativi.
- Definisci gli obiettivi: collabora con queste parti interessate per definire e documentare gli obiettivi del carico di lavoro. Considera gli aspetti critici relativi alle prestazioni dei carichi di lavoro, come il throughput, i tempi di risposta e i costi, nonché gli obiettivi aziendali, come la soddisfazione degli utenti.
- Esamina le best practice di settore: esamina le best practice del settore per individuare i KPI pertinenti in linea con gli obiettivi del carico di lavoro.
- Individua le metriche: identifica le metriche in linea con gli obiettivi del carico di lavoro e in grado di aiutarti a misurare prestazioni e obiettivi aziendali. Stabilisci i KPI in base a queste metriche, ad esempio le misurazioni del tempo medio di risposta o del numero di utenti simultanei.
- Definisci e documenta i KPI: utilizza le best practice del settore e gli obiettivi del carico di lavoro per fissare i valori dei KPI del carico di lavoro. Utilizza queste informazioni per impostare soglie dei KPI per livello di gravità o allarme. Identifica e documenta il rischio e l'impatto in caso di mancato raggiungimento del KPI.
- Implementa il monitoraggio: utilizza strumenti di monitoraggio, come Amazon CloudWatch o AWS Config, per la raccolta di metriche e la misurazione dei KPI.
- Comunica visivamente i KPI: utilizza strumenti del pannello di controllo, come Amazon Quick, per visualizzare e comunicare i KPI alle parti interessate.
- Analizza e ottimizza: esamina e analizza in modo regolare i parametri per individuare le aree del carico di lavoro da migliorare. Collabora con le parti interessate per implementare tali miglioramenti.
- Riesamina e perfeziona: rivedi con regolarità metriche e KPI per valutare la loro efficacia, soprattutto in caso di modifica di obiettivi aziendali o prestazioni del carico di lavoro.

Risorse

Documenti correlati:

- CloudWatch documentation
- Monitoring, Logging, and Performance AWS Partners
- AWS observability tools
- The Importance of Key Performance Indicators (KPIs) for Large-Scale Cloud Migrations
- How to track your cost optimization KPIs with the KPI Dashboard
- Documentazione di X-Ray
- Using Amazon CloudWatch dashboards
- KPI di Quick

Video correlati:

- AWS re:Invent 2023 - Optimize cost and performance and track progress toward mitigation
- AWS re:Invent 2023 - Manage resource lifecycle events at scale with AWS Health
- AWS re:Invent 2023 - Performance & efficiency at Pinterest: Optimizing the latest instances
- AWS re:Invent 2022 - AWS optimization: Actionable steps for immediate results
- AWS re:Invent 2023 - Building an effective observability strategy
- AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS
- AWS re:Invent 2023 - Scaling on AWS for the first 10 million users
- AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance
- Creating an Effective Metrics Strategy for Your Business | AWS Events

Esempi correlati:

- Creazione di un pannello di controllo con Quick

PERF05-BP02 Uso di soluzioni di monitoraggio per comprendere le aree in cui le prestazioni sono più critiche

Comprendi e identifica le aree in cui l'aumento delle prestazioni del carico di lavoro determinerà un impatto positivo sull'efficienza o sull'esperienza del cliente. Ad esempio, un sito web che ha

una grande quantità di interazione con i clienti può trarre vantaggio dall'utilizzo dei servizi edge per spostare la distribuzione di contenuti più vicino ai clienti.

Anti-pattern comuni:

- Si ritiene che i parametri di calcolo standard, ad esempio l'utilizzo della CPU o il carico della memoria, siano sufficienti per rilevare problemi di prestazioni.
- Utilizzo solo dei parametri predefiniti registrati dal software di monitoraggio selezionato.
- Revisione dei parametri solo quando c'è un problema.

Vantaggi dell'adozione di questa best practice: l'individuazione delle aree critiche delle prestazioni consente ai proprietari del carico di lavoro di monitorare i KPI e dare priorità ai miglioramenti ad alto impatto.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Configura il tracciamento end-to-end per identificare gli schemi di traffico, la latenza e le aree con prestazioni critiche. Monitora gli schemi di accesso ai dati per query lente o dati scarsamente frammentati e partizionati. Identifica le aree vincolate del carico di lavoro utilizzando test o monitoraggio del carico.

Aumenta l'efficienza delle prestazioni esaminando l'architettura, gli schemi di traffico e gli schemi di accesso ai dati e identifica la latenza e i tempi di elaborazione. Identifica i potenziali colli di bottiglia che potrebbero influire sull'esperienza del cliente man mano che il carico di lavoro aumenta. Dopo aver identificato queste aree, individua quale soluzione puoi implementare per evitare tali problemi di prestazioni.

Passaggi dell'implementazione

- Configura il monitoraggio end-to-end per acquisire tutti i componenti e i parametri del carico di lavoro. Ecco alcuni esempi di soluzioni di monitoraggio su AWS.

Servizio	Dove usarlo
Monitoraggio degli utenti reali di Amazon CloudWatch (RUM)	Per acquisire i parametri delle prestazioni delle applicazioni da sessioni lato client e frontend di utenti reali.

Servizio	Dove usarlo
AWS X-Ray	Per tenere traccia del traffico nei livelli dell'applicazione e identificare la latenza tra componenti e dipendenze. Utilizza le mappe del servizio X-Ray per osservare le relazioni e la latenza tra i componenti del carico di lavoro.
Informazioni dettagliate sulle prestazioni del servizio Amazon Relational Database	Per osservare i parametri delle prestazioni del database e identificare le prestazioni da migliorare.
Monitoraggio avanzato di Amazon RDS	Per osservare i parametri delle prestazioni del sistema operativo del database.
Amazon DevOps Guru	Per rilevare modelli operativi anomali in modo da poter identificare i problemi operativi prima che abbiano un impatto sui clienti.

- Esegui i test per generare parametri, identificare schemi di traffico, colli di bottiglia e aree con prestazioni critiche. Ecco alcuni esempi di come eseguire i test:
  - Configura i canary di CloudWatch Synthetic per simulare le attività degli utenti basate sul browser in modo programmatico utilizzando espressioni della frequenza o processi CRON di Linux per generare parametri coerenti nel tempo.
  - Usa la soluzione Test di carico distribuito di AWS per generare picchi di traffico o testare il carico di lavoro al tasso di crescita previsto.
- Valuta parametri e dati di telemetria per identificare le aree critiche delle prestazioni. Esamina queste aree con il tuo team per determinare il monitoraggio e le soluzioni per evitare i colli di bottiglia.
- Sperimenta i miglioramenti delle prestazioni e valuta tali modifiche con i dati. Ad esempio, puoi utilizzare CloudWatch Evidently per testare nuovi miglioramenti e gli impatti in termini di prestazioni sul tuo carico di lavoro.

Risorse

Documenti correlati:

- What's new in AWS Observability at re:Invent 2023
- Amazon Builders' Library
- Documentazione di X-Ray
- Amazon CloudWatch RUM
- Amazon DevOps Guru

Video correlati:

- AWS re:Invent 2023 - [LAUNCH] Application monitoring for modern workloads
- AWS re:Invent 2023 - Implementing application observability
- AWS re:Invent 2023 - Building an effective observability strategy
- AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS
- AWS re:Invent 2022 - AWS optimization: Actionable steps for immediate results
- AWS re:Invent 2022 - The Amazon Builders' Library: 25 years of Amazon operational excellence
- AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance
- Visual Monitoring of Applications with Amazon CloudWatch Synthetics

Esempi correlati:

- Misurazione dei tempi di caricamento delle pagine con Amazon CloudWatch Synthetics
- Client Web Amazon CloudWatch RUM
- SDK X-Ray per Python
- Test del carico distribuito su AWS

PERF05-BP03 Definizione di un processo per migliorare le prestazioni del carico di lavoro

Definisci un processo per valutare i nuovi servizi, i modelli di progettazione, i tipi di risorse e le configurazioni man mano che diventano disponibili. Ad esempio, esegui test delle prestazioni esistenti sulle nuove offerte di istanze per determinare il loro potenziale per migliorare il carico di lavoro.

Anti-pattern comuni:

- Si ritiene che l'architettura corrente diventi statica e non venga aggiornata nel corso del tempo.

- Introduzione di modifiche all'architettura nel tempo senza dei parametri che le giustifichino.

Vantaggi dell'adozione di questa best practice: definire un processo per apportare modifiche all'architettura consente ai dati raccolti di influenzare la progettazione del carico di lavoro nel corso del tempo.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Le prestazioni del carico di lavoro presentano alcuni vincoli principali. Documentali, in modo da sapere quali tipi di innovazione potrebbero migliorare le prestazioni del carico di lavoro. Utilizza queste informazioni quando vieni a conoscenza di nuovi servizi o tecnologie, man mano che si rendono disponibili, in modo da identificare le soluzioni per ovviare ai vincoli o ai colli di bottiglia.

Determina i principali vincoli riguardanti le prestazioni del carico di lavoro. Documenta i vincoli prestazionali del carico di lavoro in modo da sapere quali tipi di innovazione potrebbero migliorare le prestazioni del carico di lavoro.

Passaggi dell'implementazione

- Individua i KPI: stabilisci i KPI in termini di prestazioni del carico di lavoro come indicato in PERF05-BP01 Individuazione degli indicatori chiave di prestazioni (KPI) per misurare l’integrità e le prestazioni del carico di lavoro per definire come base il carico di lavoro.
- Implementa il monitoraggio: sfrutta gli strumenti di osservabilità AWS per raccogliere metriche delle prestazioni e misurare i KPI.
- Effettua analisi: conduci anali approfondite per individuare le aree (come la configurazione e il codice applicativo) del carico di lavoro con prestazioni insufficienti, come indicato in PERF05-BP02 Uso di soluzioni di monitoraggio per comprendere le aree in cui le prestazioni sono più critiche. Usa i tuoi strumenti di analisi e prestazioni per individuare la strategia di miglioramento delle prestazioni.
- Convalida i miglioramenti: utilizza gli ambienti sandbox o di preproduzione per convalidare l'efficacia della strategia di miglioramento.
- Implementa le modifiche: implementa le modifiche nella produzione e monitora in modo continuo le prestazioni del carico di lavoro. Documenta i miglioramenti e comunica i risultati alle parti interessate.
- Riesamina e perfeziona: rivedi con regolarità il processo di miglioramento delle prestazioni per individuare le aree di miglioramento.

Risorse

Documenti correlati:

- AWS Blog
- Novità di AWS
- AWS Skill Builder

Video correlati:

- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2023 - Optimize cost and performance and track progress toward mitigation
- AWS re:Invent 2022 - AWS optimization: Actionable steps for immediate results
- AWS re:Invent 2022 - Optimize your AWS workloads with best-practice guidance

Esempi correlati:

- GitHub AWS

PERF05-BP04 Load Esegui un test del tuo carico di lavoro

Esegui il test del carico di lavoro per verificare che sia in grado di gestire il carico di produzione e individuare eventuali colli di bottiglia nelle prestazioni.

Anti-pattern comuni:

- Test delle singole parti del carico di lavoro, ma non dell'intero carico di lavoro.
- Test di carico eseguito su un'infrastruttura diversa dall'ambiente di produzione.
- Test di carico eseguiti solo per il carico previsto e non oltre, per prevedere dove si potrebbero riscontrare problemi futuri.
- Esegui test di carico senza consultare la Amazon EC2 Testing Policy e inviare un modulo di invio di eventi simulati. Ciò comporta la mancata esecuzione del test, in quanto sembra un evento. denial-of-service

Vantaggi dell'adozione di questa best practice: misurando le prestazioni in un test di carico, potrai vedere dove avrà luogo l'impatto con l'aumento del carico. In questo modo puoi anticipare le modifiche necessarie prima che influiscano sul carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Il test di carico nel cloud è un processo volto a misurare le prestazioni del carico di lavoro in condizioni realistiche e con il carico degli utenti previsto. Questo processo prevede il provisioning di un ambiente cloud simile a quello di produzione, l'utilizzo di strumenti di test di carico per generare il carico e l'analisi dei parametri per valutare la capacità del carico di lavoro di gestire un carico realistico. Occorre eseguire i test di carico tramite versioni sintetiche o purificate dei dati di produzione (rimuovendo le informazioni sensibili o che permettono l'identificazione degli utenti). Eseguite automaticamente i test di carico come parte della vostra pipeline di distribuzione e confrontate i risultati con soglie e soglie predefiniteKPIs. Questo processo ti consente di ottenere le prestazioni richieste.

Passaggi dell'implementazione

- Definisci gli obiettivi dei test: individua gli aspetti in termini di prestazione del carico di lavoro da valutare, come il throughput e il tempo di risposta.
- Seleziona uno strumento di test: scegli e configura lo strumento di test più adatto al carico di lavoro.
- Configura l'ambiente: configura l'ambiente di test in base al tuo ambiente di produzione. Puoi utilizzare AWS i servizi per eseguire ambienti su scala di produzione per testare la tua architettura.
- Implementa il monitoraggio: utilizza strumenti di monitoraggio come Amazon CloudWatch per raccogliere metriche tra le risorse della tua architettura. Puoi anche raccogliere e pubblicare metriche personalizzate.
- Definisci gli scenari definisci scenari e parametri del test di carico (come la durata del test e il numero di utenti).
- Esegui test di carico: effettua scenari di test su vasta scala. Approfittane Cloud AWS per testare il tuo carico di lavoro e scoprire dove non riesce a scalare o se è scalabile in modo non lineare. Ad esempio, usa le istanze spot per generare carichi a costi ridotti e rilevare i colli di bottiglia prima che si verifichino in produzione.
- Analizza i risultati dei test: analizza i risultati per individuare colli di bottiglia delle prestazioni e aree di miglioramento.

- Documenta e condividi gli esiti: documenta esiti e raccomandazioni e crea report al riguardo. Condividi queste informazioni con le parti interessate per aiutarle a prendere decisioni informate sulle strategie di ottimizzazione delle prestazioni.
- Effettua iterazioni continue: esegui con regolarità i test di carico, specie dopo una modifica o un aggiornamento del sistema.

Risorse

Documenti correlati:

- Amazon CloudWatch RUM
- Amazon CloudWatch Synthetics
- Test di carico distribuito su AWS

Video correlati:

- AWS Summit ANZ 2023: accelera con fiducia grazie ai test di carico AWS distribuiti
- AWS re:Invent 2022: scalabile AWS per i primi 10 milioni di utenti
- Soluzione con AWS soluzioni: test di carico distribuiti

- AWS re:Invent 2021 - Ottimizza le applicazioni attraverso approfondimenti sugli utenti finali con Amazon CloudWatch RUM
- Demo di Amazon CloudWatch Synthetics

Esempi correlati:

- Test di carico distribuito su AWS

PERF05-BP05 Uso dell'automazione per risolvere in modo proattivo i problemi relativi alle prestazioni

Utilizza indicatori chiave di prestazioni (KPI), in combinazione con sistemi di monitoraggio e allarmi, per risolvere in modo proattivo i problemi correlati alle prestazioni.

Anti-pattern comuni:

- Solo il personale operativo è autorizzato ad apportare modifiche operative al carico di lavoro.

- Tutti gli allarmi giungono direttamente al team operativo senza alcuna correzione proattiva.

Vantaggi dell'adozione di questa best practice: la correzione proattiva delle azioni di allarme consente al personale di supporto di concentrarsi sugli elementi non attivabili in automatico. In questo modo, il personale operativo non viene sovraccaricato da tutti gli allarmi e si concentra, invece, solo sugli allarmi critici.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Laddove possibile, utilizza gli allarmi per attivare operazioni automatizzate per risolvere i problemi. Se non è possibile rispondere in modo automatizzato, inoltra l'allarme a chi può intervenire. Ad esempio, puoi implementare un sistema in grado di prevedere i valori attesi per gli indicatori chiave di prestazioni (KPI) e di inviare allarmi qualora essi oltrepassino determinate soglie, oppure uno strumento che arresta o esegue in automatico il rollback delle implementazioni in caso di discostamento dei KPI dai valori attesi.

Implementa processi che forniscono visibilità sulle prestazioni durante l'esecuzione del carico di lavoro. Crea pannelli di controllo del monitoraggio e stabilisci norme di riferimento per le aspettative in termini di prestazioni, per determinare se il carico di lavoro presenta prestazioni ottimali.

Passaggi dell'implementazione

- Identifica il flusso di correzione: individua e comprendi il problema delle prestazioni risolvibile automaticamente. Utilizza soluzioni di monitoraggio AWS come Amazon CloudWatch o AWS X-Ray per comprendere meglio la causa principale del problema.
- Definisci il processo di automazione: crea un processo di risoluzione dettagliato utilizzabile per risolvere in automatico il problema.
- Configura l'evento di avvio: configura l'evento per l'avvio automatico del processo di risoluzione. Ad esempio, è possibile definire un trigger per riavviare automaticamente un'istanza quando raggiunge una determinata soglia di utilizzo della CPU.
- Automatizza la correzione: utilizza i servizi e le tecnologie AWS per automatizzare il processo di risoluzione. Ad esempio, AWS Systems Manager Automation fornisce un modo sicuro e scalabile per automatizzare il processo di risoluzione. Assicurati di utilizzare la logica di risoluzione automatica per annullare le modifiche se non risolvono correttamente il problema.
- Testa il flusso di lavoro: esegui il test del processo di risoluzione automatizzato in un ambiente di preproduzione.

- Implementa il flusso di lavoro: implementa la risoluzione automatizzata nell'ambiente di produzione.
- Sviluppa un playbook: predisponi e documenta un playbook che delinei le fasi del piano di risoluzione, inclusi eventi di avvio, logica di risoluzione e azioni intraprese. Assicurati di fornire la giusta preparazione alle parti interessate in modo che possano rispondere efficacemente agli eventi di risoluzione automatizzati.
- Esamina e perfeziona: valuta con regolarità l'efficacia del flusso di lavoro di risoluzione automatizzato. Modifica gli eventi di avvio e la logica di risoluzione, se necessario.

Risorse

Documenti correlati:

- CloudWatch Documentation
- Monitoraggio, registrazione di log e prestazioni: partner AWS Partner Network
- Documentazione di X-Ray
- Using Alarms and Alarm Actions in CloudWatch
- Build a Cloud Automation Practice for Operational Excellence: Best Practices from AWS Managed Services
- Automate your Amazon Redshift performance tuning with automatic table optimization

Video correlati:

- AWS re:Invent 2023 - Strategies for automated scaling, remediation, and smart self-healing
- AWS re:Invent 2023 - [LAUNCH] Application monitoring for modern workloads
- AWS re:Invent 2023 - Implementing application observability
- AWS re:Invent 2021 - Intelligently automating cloud operations
- AWS re:Invent 2022 - Setting up controls at scale in your AWS environment
- AWS re:Invent 2022 - Automating patch management and compliance using AWS
- AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance
- AWS re:Invent 2023 - Take a load off: Diagnose & resolve performance issues with Amazon RDS
- AWS re:Invent 2021 -{New Launch} Automatically detect and resolve issues with Amazon DevOps Guru
- AWS re:Invent 2023 - Centralize your operations

Esempi correlati:

- CloudWatch Logs Customize Alarms

PERF05-BP06 Conserva il carico di lavoro e i servizi up-to-date

Resta up-to-date su nuovi servizi e funzionalità cloud per adottare funzionalità efficienti, rimuovere problemi e migliorare l'efficienza complessiva delle prestazioni del tuo carico di lavoro.

Anti-pattern comuni:

- Si ritiene che l'architettura corrente diventi statica e non venga aggiornata nel corso del tempo.
- Non si dispone di sistemi né si esegue regolarmente una valutazione per la compatibilità di software e pacchetti aggiornati con il carico di lavoro.

Vantaggi derivanti dall'adozione di questa best practice: stabilendo un processo per rimanere aggiornato up-to-date su nuovi servizi e offerte, puoi adottare nuove funzionalità e funzionalità, risolvere problemi e migliorare le prestazioni del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Valuta i modi per migliorare le prestazioni man mano che nuovi servizi, modelli di progettazione e funzionalità di prodotti diventano disponibili. Determina in che modo possono migliorare le prestazioni o aumentare l'efficienza del carico di lavoro tramite valutazioni, discussioni interne o analisi esterne. Definisci un processo per valutare gli aggiornamenti, le nuove funzionalità e i servizi pertinenti per il tuo carico di lavoro. Ad esempio, crea un proof of concept che utilizza le nuove tecnologie o consultati con un gruppo interno. Quando provi nuove idee o servizi, esegui test delle prestazioni per misurare l'impatto sulle prestazioni del carico di lavoro.

Passaggi dell'implementazione

- Esegui l'inventario del tuo carico di lavoro: esegui l'inventario di software e architettura del carico di lavoro e identifica i componenti da aggiornare.
- Identifica le origini dell'aggiornamento: identifica novità e origini dell'aggiornamento relative ai componenti del carico di lavoro. Ad esempio, puoi iscriverti al AWS blog What's New at per i prodotti che corrispondono al tuo componente di carico di lavoro. Puoi iscriverti al RSS feed o gestire le tue iscrizioni e-mail.

- Definisci un programma di aggiornamento: definisci un programma per valutare nuovi servizi e funzionalità per il tuo carico di lavoro.
  - Puoi utilizzare AWS Systems Manager Inventory per raccogliere i metadati del sistema operativo (OS), delle applicazioni e delle istanze dalle tue EC2 istanze Amazon e capire rapidamente quali istanze eseguono il software e le configurazioni richieste dalla tua politica software e quali istanze devono essere aggiornate.
- Valuta il nuovo aggiornamento: individua le modalità di aggiornamento dei componenti del carico di lavoro. Sfrutta l'agilità del cloud per testare in modo semplice e rapido il modo in cui le nuove funzionalità possono migliorare il carico di lavoro per ottenere efficienza delle prestazioni.
- Utilizza l'automazione: sfrutta l'automazione del processo di aggiornamento per ridurre il livello di impegno per implementare le nuove funzionalità e limitare gli errori causati dai processi manuali.
  - Puoi utilizzare CI/CD per aggiornare AMIs automaticamente le immagini dei container e altri elementi relativi alla tua applicazione cloud.
  - È possibile utilizzare strumenti come AWS Systems Manager Patch Manager per automatizzare il processo di aggiornamento del sistema e pianificare l'attività utilizzando le finestre di manutenzione di AWS Systems Manager.
- Documenta il processo: documenta il tuo processo di valutazione di aggiornamenti e nuovi servizi. Fornisci ai proprietari il tempo e lo spazio necessari per ricercare, testare, sperimentare e convalidare aggiornamenti e nuovi servizi. Fate riferimento ai requisiti aziendali documentati e aiutateci KPIs a stabilire le priorità degli aggiornamenti che avranno un impatto positivo sull'azienda.

Risorse

Documenti correlati:

- Blog AWS
- Cosa c'è di nuovo con AWS
- Implementazione di up-to-date immagini con pipeline automatizzate di EC2 Image Builder

Video correlati:

- AWS RE:InForce 2022 - Automatizzazione della gestione e della conformità delle patch utilizzando AWS
- All Things Patch: | Eventi AWS Systems ManagerAWS

Esempi correlati:

- Inventory and Patch Management
- One Observability Workshop

PERF05-BP07 Analisi dei parametri a intervalli regolari

Nell'ambito della manutenzione ordinaria o in risposta a eventi o incidenti, esamina i parametri raccolti. Stabilisci quali di questi parametri sono fondamentali per risolvere i problemi e quali altri parametri aggiuntivi, se monitorati, possono contribuire a identificare, affrontare o prevenire i problemi.

Anti-pattern comuni:

- Si lascia che i parametri rimangano in uno stato di allarme per un lungo periodo di tempo.
- Creazione di allarmi non utilizzabili da un sistema di automazione.

Vantaggi dell'adozione di questa best practice: esamina in modo continuo i parametri raccolti per verificare che identifichino, risolvano o prevengano adeguatamente i problemi. I parametri possono anche diventare obsoleti se lasciati in uno stato di allarme per un lungo periodo di tempo.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Migliora continuamente la raccolta e il monitoraggio dei parametri. Nell'ambito della risposta a incidenti ed eventi, valuta quali parametri sono stati utili per affrontare il problema e quali sarebbero stati utili ma non sono attualmente misurati. Questo metodo ti aiuterà a migliorare la qualità dei parametri raccolti, in modo da prevenire o risolvere in modo più rapido gli incidenti futuri.

Nell'ambito della risposta a incidenti ed eventi, valuta quali parametri sono stati utili per affrontare il problema e quali sarebbero stati utili ma non sono attualmente misurati. Queste considerazioni ti aiuteranno a migliorare la qualità dei parametri raccolti, così da prevenire o risolvere più rapidamente gli incidenti futuri.

Passaggi dell'implementazione

- Definisci metriche: stabilisci metriche in termini di prestazioni critiche da monitorare, allineate all'obiettivo del carico di lavoro, incluse metriche quali il tempo di risposta e l'utilizzo delle risorse.

- Stabilisci una base: imposta un valore di base e auspicabile per ciascuna metrica. La base e deve fornire i punti di riferimento per identificare deviazioni o anomalie.
- Imposta una cadenza: imposta una cadenza (ad esempio, settimanale o mensile) per rivedere le metriche più critiche.
- Identifica i problemi di prestazioni: durante ogni revisione, valuta tendenze e deviazione dai valori di base. Cerca eventuali colli di bottiglia o anomalie nelle prestazioni. Per i problemi identificati, esegui un'analisi approfondita delle cause principali per comprendere il motivo più importante alla base del problema.
- Individua le azioni correttive: utilizza l'analisi per identificare le azioni correttive, come l'ottimizzazione dei parametri, la correzione di bug e il dimensionamento delle risorse.
- Documenta gli esiti: documenta gli esiti, compresi i problemi identificati, le cause principali e le azioni correttive.
- Itera migliora: valuta e migliora continuamente il processo di revisione delle metriche. Usa le indicazioni apprese dalla revisione precedente per migliorare il processo nel tempo.

Risorse

Documenti correlati:

- CloudWatch Documentation
- Collect metrics and logs from Amazon EC2 Instances and on-premises servers with the CloudWatch Agent
- Query your metrics with CloudWatch Metrics Insights
- Monitoraggio, registrazione di log e prestazioni: partner AWS Partner Network
- Documentazione di X-Ray

Video correlati:

- AWS re:Invent 2022 - Setting up controls at scale in your AWS environment
- AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance
- AWS re:Invent 2023 - Building an effective observability strategy
- AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS
- AWS re:Invent 2023 - Take a load off: Diagnose & resolve performance issues with Amazon RDS

Esempi correlati:

- Creazione di un pannello di controllo con Quick
- CloudWatch Dashboards

Ottimizzazione dei costi

Il pilastro dell'ottimizzazione dei costi include la possibilità di eseguire sistemi per offrire valore aggiunto al prezzo più basso. Le linee guida con le prescrizioni sull'implementazione sono disponibili nel whitepaper sul pilastro dell'ottimizzazione dei costi.

Aree delle best practice

- Implementazione della gestione finanziaria del cloud
- Comprensione delle spese e dell'utilizzo
- Risorse convenienti in termini di costo
- Gestione delle risorse di domanda e offerta
- Ottimizzazione nel tempo

Implementazione della gestione finanziaria del cloud

Domanda

- COST 1. Come implementi la gestione finanziaria nel cloud?

COST 1. Come implementi la gestione finanziaria nel cloud?

L'implementazione della gestione finanziaria del cloud aiuta le organizzazioni a conseguire un valore aggiunto e il successo finanziario ottimizzando i costi e l'utilizzo e dimensionando le risorse in AWS.

Best practice

- COST01-BP01 Stabilire la titolarità dell'ottimizzazione dei costi
- COST01-BP02 Definizione di una partnership tra team finanziari e tecnologici
- COST01-BP03 Definizione di budget e previsioni per il cloud
- COST01-BP04 Implementazione della consapevolezza dei costi nei processi dell'organizzazione
- COST01-BP05 Invio di report e notifiche sull’ottimizzazione dei costi

- COST01-BP06 Monitoraggio proattivo dei costi
- COST01-BP07 Resta aggiornato up-to-date sulle nuove release di servizio
- COST01-BP08 Creazione di una cultura consapevole dei costi
- COST01-BP09 Quantifica il valore aziendale grazie all'ottimizzazione dei costi

COST01-BP01 Stabilire la titolarità dell'ottimizzazione dei costi

Crea un team (Cloud Business Office, Cloud Center of Excellence o FinOps team) responsabile della creazione e del mantenimento della consapevolezza dei costi in tutta l'organizzazione. Il responsabile dell'ottimizzazione dei costi può essere un individuo o un team (sono necessarie persone provenienti da team finanziari, tecnologici e aziendali) che ha una comprensione dell'intera organizzazione e degli aspetti finanziari legati al cloud.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Si tratta dell'introduzione di una funzione o di un team di Cloud Business Office (CBOCCOE) o Cloud Center of Excellence () responsabile della creazione e del mantenimento di una cultura della consapevolezza dei costi nel cloud computing. Questa funzione può essere una figura professionale già in organico, un team all'interno della tua organizzazione o un nuovo team di parti interessate chiave dei settori finanza, tecnologia e organizzazione provenienti da tutta l'azienda.

La funzione (individuo o team) stabilisce le priorità e dedica la parte prevista del proprio tempo alle attività di gestione e ottimizzazione dei costi. In un'organizzazione di dimensioni ridotte, la quantità di tempo dedicata dalla funzione potrebbe essere inferiore rispetto a quella dedicata da una funzione a tempo pieno in un'azienda di dimensioni maggiori.

La funzione richiede un approccio multidisciplinare, con capacità di gestione dei progetti, data science, analisi finanziaria e sviluppo di software o infrastruttura. Può migliorare l'efficienza del carico di lavoro eseguendo ottimizzazioni dei costi all'interno di tre diversi tipi di responsabilità:

- Centralizzato: tramite team designati come FinOps team, team Cloud Financial Management (CFM), Cloud Business Office (CBO) o Cloud Center of Excellence (CCoE), i clienti possono progettare e implementare meccanismi di governance e promuovere le migliori pratiche a livello aziendale.
- Team decentralizzati: influenzano i team tecnologici per ottimizzare i costi.

- Team ibridi: una combinazione di team centralizzati e decentralizzati può collaborare per eseguire l'ottimizzazione dei costi.

La funzione può essere valutata in base alla sua capacità di eseguire e conseguire risultati rispetto agli obiettivi di ottimizzazione dei costi (ad esempio in base a metriche di efficienza dei carichi di lavoro).

Un fattore chiave per il successo di questa funzione è la disponibilità di sponsorizzazione da parte del management. Lo sponsor deve essere un sostenitore del consumo efficiente del cloud e fornire alla funzione un supporto in caso di escalation, per garantire che le attività di ottimizzazione dei costi vengano trattate con il livello di priorità definito dall'organizzazione. In caso contrario, le linee guida possono essere ignorate e non verrà data priorità alle opportunità di riduzione dei costi. Insieme, lo sponsor e il team dell'organizzazione possono aiutare a utilizzare il cloud in modo efficiente e generare valore aziendale.

Se disponi del piano di supporto Business Enterprise-On-Ramp o Enterprise e hai bisogno di aiuto per creare questo team o questa funzione, contatta gli esperti di Cloud Financial Management (CFM) tramite il tuo account team.

Passaggi dell'implementazione

- Definizione dei membri chiave: tutte le parti rilevanti della tua organizzazione devono contribuire ed essere interessate alla gestione dei costi. I team comuni all'interno delle organizzazioni includono in genere: responsabili finanziari, proprietari delle applicazioni o dei prodotti, team di gestione e tecnici (DevOps). Alcuni soggetti sono impegnati a tempo pieno (ad esempio quelli di tipo finanziario o tecnico), mentre altri sono coinvolti periodicamente secondo necessità. Gli individui o i team che si esibiscono CFM necessitano delle seguenti competenze:
  - Sviluppo software: competenze inerenti allo sviluppo di software, in caso di sviluppo di script e funzioni di automazione.
  - Progettazione dell'infrastruttura: per implementare script, automatizzare processi e comprendere in che modo vengono allocati risorse e servizi.
  - Acume operativo: CFM consiste nell'operare sul cloud in modo efficiente misurando, monitorando, modificando, pianificando e scalando l'uso efficiente del cloud.
- Definizione di obiettivi e metriche: la funzione deve fornire valore all'organizzazione in modi diversi. Questi obiettivi sono definiti e si evolvono continuamente con l'evolversi dell'organizzazione. Tra le attività più comuni figurano la creazione e l'esecuzione di programmi di formazione sull'ottimizzazione dei costi in tutta l'organizzazione, lo sviluppo di standard a livello aziendale,

come monitoraggio ed elaborazione di report per l'ottimizzazione dei costi, e la definizione degli obiettivi di ottimizzazione dei carichi di lavoro. Inoltre, è necessario comunicare regolarmente all'organizzazione la relativa capacità di ottimizzazione dei costi.

È possibile definire indicatori chiave di prestazione basati sul valore o sul costo (). KPIs Quando si definisce ilKPIs, è possibile calcolare il costo previsto in termini di efficienza e risultati aziendali attesi. Basate sul valore, collegate KPIs le metriche di costo e utilizzo ai fattori di valore aziendale e aiutate a razionalizzare le variazioni di spesa. AWS Il primo passo per ricavare valori basati sul valore KPIs consiste nel lavorare insieme, a livello interorganizzativo, per selezionare e concordare un set standard di. KPIs

- Definizione di una cadenza regolare: il gruppo (team finanziario, tecnologico e aziendale) deve riunirsi regolarmente per rivedere le metriche e gli obiettivi. Una periodicità tipica implica la revisione dello stato dell'organizzazione, la revisione dei programmi attualmente in esecuzione e la revisione dei parametri finanziari e di ottimizzazione generali. Quindi, per i carichi di lavoro chiave, è opportuno elaborare report più dettagliati.

  Durante queste riunioni periodiche è possibile analizzare l'efficienza (costo) dei carichi di lavoro e il risultato aziendale. Ad esempio, un incremento del 20% dei costi di un carico di lavoro potrebbe essere determinato dall'aumento dell'utilizzo da parte dei clienti. In questo caso, l'incremento del 20% dei costi può essere interpretato come investimento. Queste chiamate a cadenza regolare possono aiutare i team a identificare i valori KPIs che danno significato all'intera organizzazione.

Risorse

Documenti correlati:

- Blog AWS CCOE
- Creating Cloud Business Office
- CCOE- Centro di eccellenza cloud

Video correlati:

- Storia di CCOE successo di Vanguard

Esempi correlati:

- Utilizzo di un Cloud Center of Excellence (CCOE) per trasformare l'intera azienda

- Costruire un CCOE programma per trasformare l'intera azienda
- 7 insidie da evitare durante la costruzione CCOE

COST01-BP02 Definizione di una partnership tra team finanziari e tecnologici

Coinvolgi i team finanziari e tecnologici nelle discussioni su costi e utilizzo in tutte le fasi del tuo percorso verso il cloud. I team si riuniscono regolarmente e discutono argomenti quali obiettivi e target organizzativi, stato attuale di costi e utilizzo e pratiche finanziarie e contabili.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I team tecnologici possono innovare più rapidamente nel cloud grazie a cicli di approvazione, approvvigionamento e implementazione dell'infrastruttura più brevi. Può trattarsi di una novità per le organizzazioni finanziarie che in precedenza erano abituate a eseguire processi dispendiosi, in termini di tempo e di risorse, per acquistare e distribuire capitale in data center e on-premises, allocando i costi solo in fase di approvazione del progetto.

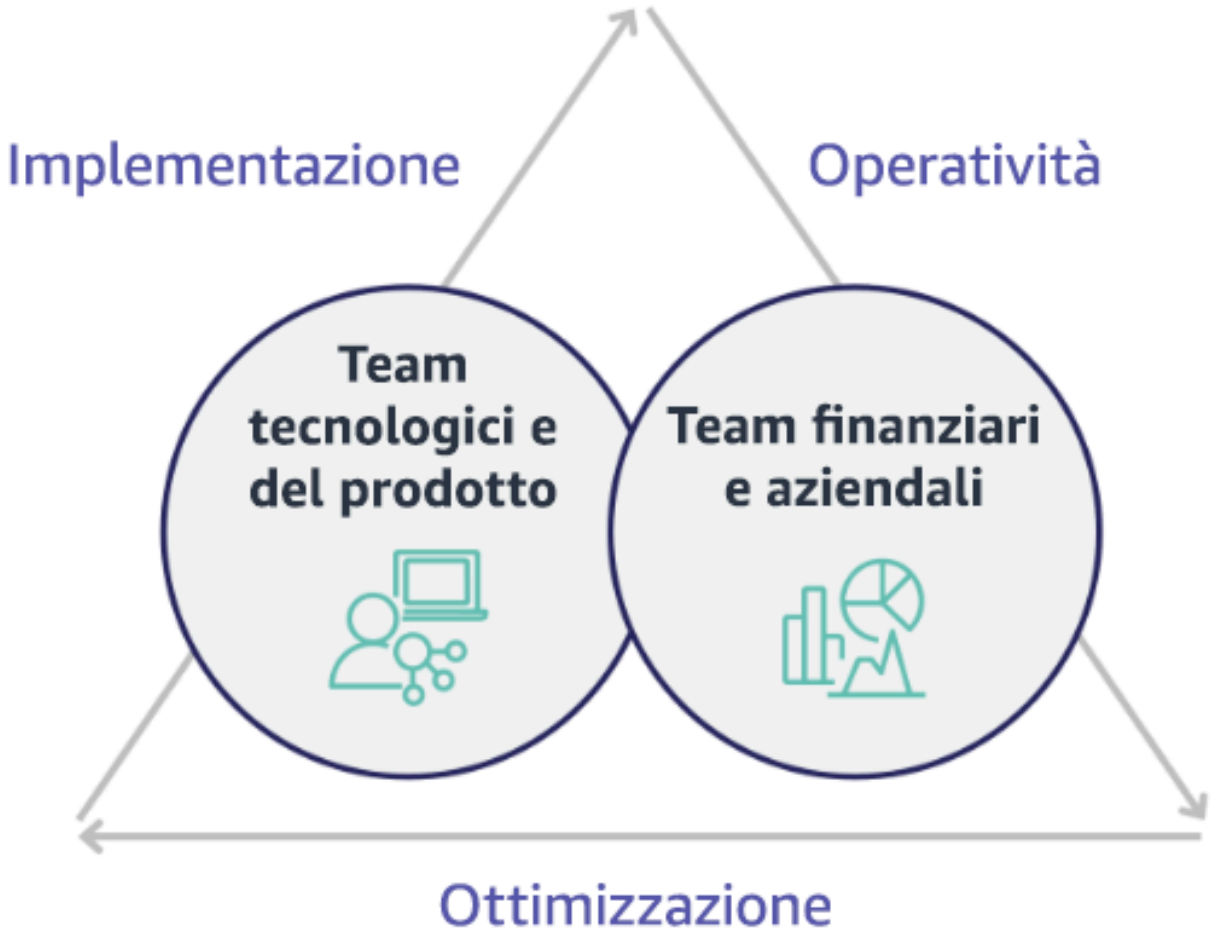
Dal punto di vista delle organizzazioni finanziarie e addette all'approvvigionamento, il processo di elaborazione del piano degli investimenti, della richiesta, dell'approvazione e dell'approvvigionamento degli investimenti e dell'installazione dell'infrastruttura fisica è stato interiorizzato e standardizzato da decenni:

- I team di progettazione o IT sono in genere i richiedenti
- I vari team finanziari fungono da approvatori e addetti all'approvvigionamento
- I team operativi assemblano, implementano e distribuiscono un'infrastruttura pronta all'uso

Con l'adozione del cloud, l'approvvigionamento e il consumo dell'infrastruttura non sono più vincolati da una catena di dipendenze. Nel modello cloud, i team tecnologici e del prodotto non sono più semplici sviluppatori, ma anche operatori e proprietari dei loro prodotti, responsabili della maggior parte delle attività storicamente associate ai team finanziari e operativi, compresi l'approvvigionamento e l'implementazione.

Quanto in realtà è necessario per il provisioning delle risorse cloud è un account e il set appropriato di autorizzazioni, elementi questi che riducono i rischi IT e finanziari. Ciò significa che ai team basta un numero ridotto di clic o chiamate API per terminare le risorse cloud non necessarie o inattive. Ciò inoltre consente ai team tecnologici di velocizzare l'innovazione, grazie all'agilità e alla capacità di potenziare e quindi ridimensionare i vari progetti sperimentali. Se da un lato la natura variabile del

consumo del cloud può influenzare la prevedibilità dal punto di vista del processo di elaborazione del piano degli investimenti e delle previsioni, il cloud fornisce alle organizzazioni la capacità di ridurre il costo del provisioning eccessivo e contemporaneamente il costo delle opportunità associato a un provisioning insufficiente di carattere conservativo.



Stabilisci una collaborazione tra le principali parti interessate finanziarie e tecnologiche per creare una conoscenza condivisa degli obiettivi organizzativi e sviluppare meccanismi che consentano il successo finanziario nel modello di spesa variabile del cloud computing. I team pertinenti all'interno della tua organizzazione devono essere coinvolti nelle discussioni su costi e utilizzo in tutte le fasi del tuo percorso verso il cloud; tra di essi vi sono:

- Responsabili finanziari: CFO, addetti ai controlli finanziari, pianificatori finanziari, analisti aziendali, addetti agli acquisti, al sourcing e alla contabilità fornitori devono comprendere il modello di consumo del cloud, le opzioni di acquisto e il processo di fatturazione mensile. I team finanziari devono collaborare con i team tecnologici per creare e divulgare a livello aziendale una narrazione

del valore IT che aiuti i team aziendali a comprendere lo stretto legame tra spesa in tecnologie e risultati aziendali. In questo modo, la spesa tecnologica viene considerata non tanto come un costo, quanto piuttosto come un vero e proprio investimento. A causa delle differenze fondamentali tra il cloud (ad esempio il tasso di variazione dell'utilizzo, il pagamento in base al consumo o a scaglioni, i modelli di prezzo e le informazioni dettagliate su fatturazione e utilizzo) e le operazioni on-premises, è essenziale che l'organizzazione finanziaria capisca in che modo l'utilizzo del cloud può influire sugli aspetti aziendali, tra cui processi di approvvigionamento, monitoraggio degli incentivi, allocazione dei costi e bilanci.

- Responsabili tecnologici: i responsabili tecnologici (inclusi i proprietari di prodotti e applicazioni) devono essere a conoscenza dei requisiti finanziari (ad esempio i vincoli di budget) e dei requisiti aziendali (ad esempio i contratti sul livello di servizio). In questo modo, il carico di lavoro può essere implementato in modo opportuno per raggiungere gli obiettivi desiderati dall'azienda.

La collaborazione tra finanza e tecnologia offre i seguenti vantaggi:

- I team finanziari e tecnologici hanno una visibilità quasi in tempo reale su costi e utilizzo.
- I team finanziari e tecnologici stabiliscono una procedura operativa standard per gestire le variazioni di spesa nel cloud.
- Le parti interessate finanziarie fungono da consulenti strategici per quanto riguarda il modo in cui il capitale viene utilizzato per acquistare sconti a fronte di impegni (ad esempio, istanze riservate o AWS Savings Plans) e il modo in cui il cloud viene utilizzato per far crescere l'organizzazione.
- I processi di approvvigionamento e di contabilità esistenti vengono applicati al cloud.
- I team finanziari e tecnologici collaborano per prevedere costi e utilizzo di AWS futuri, al fine di allineare e sviluppare i budget aziendali.
- La comunicazione all'interno dell'organizzazione migliora attraverso un linguaggio condiviso e una comprensione comune dei concetti finanziari.

Altre parti interessate all'interno della tua organizzazione che devono essere coinvolti nelle discussioni su costi e utilizzo includono:

- Proprietari delle business unit: i proprietari delle business unit devono comprendere il modello aziendale del cloud in modo da indirizzare l'operato delle business unit e di tutta l'azienda. Questa conoscenza del cloud è fondamentale quando è necessario prevedere la crescita e l'utilizzo del carico di lavoro, ma anche quando si valutano le diverse opzioni di acquisto, come le istanze riservate o i Savings Plans.

- Team di progettazione: lo sviluppo di una partnership tra team finanziari e tecnologici è essenziale per la creazione di una cultura consapevole dei costi che incoraggi il coinvolgimento degli ingegneri nel Cloud Financial Management (CFM). Uno dei problemi comuni dei professionisti del CFM o delle operazioni e dei team finanziari è far capire agli ingegneri l'attività nel cloud nel suo complesso e implementare le azioni consigliate.
- Terze parti: se la tua organizzazione si avvale di terze parti (ad esempio, consulenti o strumenti), assicurati che esse siano allineate ai tuoi obiettivi finanziari e possano dimostrare sia l'allineamento, tramite i loro modelli di coinvolgimento, sia il ritorno sull'investimento (ROI). In genere, le terze parti contribuiscono alla creazione di report e all'analisi di eventuali carichi di lavoro da esse gestiti, e forniscono anche l'analisi dei costi relativi ai carichi di lavoro da esse progettati.

L'implementazione del CFM e il conseguimento dei risultati richiedono la stretta collaborazione tra team finanziari, tecnologici e aziendali, nonché un cambiamento nel modo in cui la spesa cloud viene comunicata e valutata all'interno dell'organizzazione. Includi i team di progettazione in modo da renderli partecipi delle discussioni su costi e utilizzi in tutte le fasi e incoraggiali ad attenersi alle best practice e ad adottare le azioni concordate.

Passaggi dell'implementazione

- Definizione dei membri chiave: verifica che tutti i membri rilevanti dei team finanziari e tecnologici partecipino alla partnership. I membri del team finanziario interessati saranno quelli che hanno a che fare con la fatturazione dei servizi cloud. In genere si tratta di CFO, addetti ai controlli finanziari, pianificatori finanziari, analisti aziendali, addetti agli acquisti e all'approvvigionamento. I membri tecnologici sono in genere i proprietari di prodotti e applicazioni, manager tecnici e rappresentanti di tutti i team che si basano sul cloud. Altri membri possono includere i responsabili di business unit, ad esempio il marketing che influenzerà l'utilizzo dei prodotti, e terze parti, come i consulenti, necessari per garantire l'allineamento agli obiettivi e meccanismi e per fornire assistenza nell'elaborazione dei report.
- Definizione degli argomenti di discussione: definisci gli argomenti comuni tra i team o che necessitano di una comprensione condivisa. Segui il costo dal momento in cui viene creato, fino al pagamento della fattura. Prendi nota di tutti i membri coinvolti e dei processi organizzativi che devono essere applicati. Comprendi ogni fase o processo e le informazioni associate, come i modelli di prezzo disponibili, i prezzi a scaglioni, i modelli di sconto, il budget e i requisiti finanziari.
- Definizione di una regolare cadenza: per creare una partnership tra team finanziari e tecnologici, definisci la periodicità delle comunicazioni per creare e gestire l'allineamento. Il gruppo deve riunirsi regolarmente in base ai propri obiettivi e parametri. Una periodicità tipica implica la revisione dello

stato dell'organizzazione, la revisione dei programmi attualmente in esecuzione e la revisione dei parametri finanziari e di ottimizzazione generali. Quindi, per i carichi di lavoro chiave, è opportuno elaborare report più dettagliati.

Risorse

Documenti correlati:

- AWS Blog delle novità

COST01-BP03 Definizione di budget e previsioni per il cloud

Adatta i processi di previsione e di budgeting organizzativi esistenti in modo che siano compatibili con la natura altamente variabile dei costi e dell'utilizzo del cloud. I processi devono essere dinamici, utilizzando algoritmi basati su tendenze o fattori chiave aziendali o una combinazione di entrambi.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Nelle tradizionali configurazioni IT on-premises, i clienti spesso devono affrontare la sfida di pianificare i costi fissi che variano solo occasionalmente, di solito con i nuovi acquisti di hardware e servizi IT per soddisfare i picchi di domanda. Cloud AWS adotta invece un approccio diverso, in cui i clienti pagano per le risorse che utilizzano in base alle loro effettive esigenze IT e aziendali. Nell'ambiente cloud, la domanda può variare su base mensile, giornaliera o persino oraria.

Il cloud offre efficienza, velocità e agilità, consolidando un modello di costo e utilizzo altamente variabile. I costi possono diminuire o talvolta aumentare in seguito all'incremento dell'efficienza dei carichi di lavoro o all'implementazione di nuovi carichi di lavoro e funzionalità. Man mano che i carichi di lavoro scalano per servire la clientela in crescita, l'utilizzo e i costi del cloud aumentano di conseguenza a causa del maggiore uso di risorse. Questa flessibilità dei servizi cloud si estende ai costi e alle previsioni, offrendo un certo grado di elasticità.

Per ottenere la pianificazione più accurata possibile, è essenziale allinearsi prontamente a queste mutevoli esigenze aziendali e ai fattori trainanti della domanda. I tradizionali processi di budget dell'organizzazione devono cambiare per far fronte a questa variabilità.

Valuta la modellazione dei costi mentre prevedi la spesa dei nuovi carichi di lavoro. La modellazione dei costi crea una comprensione di base dei costi del cloud previsti che ti consente di calcolare

il costo totale di proprietà (TCO), il ritorno sull'investimento (ROI) e altri dati finanziari nonché di stabilire obiettivi e aspettative con le parti interessate e identificare le opportunità di ottimizzazione dei costi.

È necessario che l'organizzazione comprenda la definizione dei costi e i raggruppamenti accettati. Il livello di dettaglio usato per le previsioni può variare in base alla struttura dell'organizzazione e ai flussi di lavoro interni. Scegli la granularità adatta ai tuoi requisiti specifici e alla configurazione dell'organizzazione. È importante comprendere a quale livello viene eseguita la previsione:

- Account di gestione o livello AWS Organizations: l'account di gestione è quello utilizzato per creare AWS Organizations. Le organizzazioni dispongono di un account di gestione in modo predefinito.
- Account collegato o membro: un account in Organizations è un Account AWS standard che contiene le tue risorse AWS e le identità che possono accedere a tali risorse.
- Ambiente: un ambiente è una raccolta di risorse AWS che eseguono una versione dell'applicazione. È possibile creare un ambiente con più account collegati o membri.
- Progetto: per progetto si intende una combinazione di obiettivi o attività prestabiliti da realizzare entro un determinato periodo di tempo È importante considerare il ciclo di vita del progetto durante la previsione.
- Servizi AWS: gruppi o categorie, come servizi di calcolo o archiviazione in cui è possibile raggruppare i servizi AWS per le previsioni.
- Raggruppamento personalizzato: puoi creare gruppi personalizzati in base alle esigenze dell'organizzazione, ad esempio business unit, centri di costo, team, tag di allocazione dei costi, categorie di costi, account collegati o una combinazione di questi.

Individua i fattori aziendali che possono influire sui costi di utilizzo e fai le previsioni per ciascuno di essi separatamente per calcolare in anticipo l'utilizzo previsto. Alcuni fattori possono essere collegati ai team IT e di prodotto dell'organizzazione. Altri fattori aziendali, come eventi di marketing, promozioni, espansioni geografiche, fusioni e acquisizioni, sono noti ai responsabili dell'area vendite, marketing e commerciale, quindi è importante collaborare e tenere conto anche di tutti questi fattori trainanti della domanda.

Puoi usare AWS Cost Explorer per elaborare previsioni basate sulle tendenze per un intervallo di tempo futuro definito in base alle spese pregresse. Il motore di previsione di AWS Cost Explorer segmenta i dati storici in base ai tipi di addebito, ad esempio le istanze riservate, e utilizza una combinazione di machine learning e modelli basati su regole per elaborare previsioni di spesa per tutti i singoli tipi di addebito.

Una volta stabilito il processo di previsione e creato i modelli, Budget AWS ti permette di impostare budget personalizzati a livello granulare, specificando periodo di tempo, ricorrenza o importo (fisso o variabile), e aggiungere filtri come servizi, Regione AWS e tag. Il budget è generalmente definito per un solo anno e rimane fisso, richiedendo il rispetto rigoroso di tutte le parti coinvolte. Al contrario, le previsioni sono più flessibili, consentono adattamenti nel corso dell'anno e forniscono proiezioni dinamiche su un periodo di uno, due o tre anni. I budget e le previsioni svolgono un ruolo determinante nella definizione delle aspettative finanziarie tra le varie parti interessate tecnologiche e aziendali. Una previsione e un'implementazione accurate rendono responsabili anche le parti interessate che sono direttamente coinvolte nella gestione dei costi di provisioning e possono aumentare la loro consapevolezza generale dei costi.

Per essere informati sulle prestazioni dei budget esistenti, puoi creare e pianificare report Budget AWS da inviare tramite e-mail alle parti interessate con cadenza regolare. Puoi anche creare avvisi di Budget AWS basati sui costi effettivi, ovvero avvisi intrinsecamente reattivi, oppure sui costi previsti, ossia avvisi che consentono di implementare tempestivamente azioni correttive a fronte di potenziali eventi di superamento dei costi. Puoi ricevere un avviso quando il costo o l'utilizzo supera un determinato livello oppure si prevede che superi l'importo definito nel budget.

Modifica i processi di budget e previsione esistenti per renderli più dinamici utilizzando gli algoritmi basati sulle tendenze con i costi storici come input e gli algoritmi basati sui fattori aziendali, ad esempio il lancio di nuovi prodotti, l'espansione regionale o i nuovi ambienti per i carichi di lavoro, ideali per un ambiente di spesa dinamico e variabile. Una volta determinata la previsione basata sulle tendenze mediante Cost Explorer o qualsiasi altro strumento, utilizza Calcolatore dei prezzi AWS per stimare il caso d'uso AWS, nonché i costi futuri, in base all'utilizzo previsto (traffico, richieste al secondo o istanze Amazon EC2 necessarie).

Controlla l'accuratezza di questa previsione perché i budget devono essere impostati sulla base di questi calcoli e queste stime. Monitora la precisione e l'efficacia delle previsioni dei costi del cloud integrate. Esamina con regolarità la spesa effettiva rispetto alla tua previsione e apporta le modifiche necessarie per ottenere una maggiore accuratezza. Controlla la varianza prevista ed esegui l'analisi della causa principale della varianza indicata per intervenire e modificare le previsioni.

Come indicato in COST01-BP02 Definizione di una partnership tra team finanziari e tecnologici, è importante promuovere partnership e cadenza tra IT, finanza e altre parti interessate per verificare che tutti utilizzino gli stessi strumenti o processi per garantire la coerenza. Nei casi in cui si rendano necessarie modifiche del budget, l'incremento della frequenza delle occasioni di contatto permette di intervenire e reagire più tempestivamente.

Passaggi dell'implementazione

- Definisci il linguaggio dei costi nell'organizzazione: crea un linguaggio AWS dei costi comune all'interno dell'organizzazione con più dimensioni e raggruppamenti. Assicurati che le parti interessate comprendano la granularità delle previsioni, i modelli di prezzo e il livello delle previsioni dei costi.
- Analizza le previsioni basate sulle tendenze: utilizza strumenti per le previsioni basate sulle tendenze, come AWS Cost Explorer e Amazon Forecast. Analizza i costi di utilizzo rispetto a più dimensioni, come servizi, account, tag e categorie di costi.
- Analizza le previsioni basate sui fattori aziendali: identifica l'impatto dei fattori aziendali sull'utilizzo del cloud e fai previsioni per ciascuno di essi separatamente per calcolare in anticipo il costo di utilizzo previsto. Collabora a stretto contatto con i responsabili delle business unit e le parti interessate per comprendere l'impatto dei nuovi fattori aziendali e calcolare le variazioni dei costi previste per definire budget accurati.
- Aggiorna i processi di previsione e budget esistenti: definisci i tuoi processi di previsione del budget in base ai metodi di previsione adottati, ad esempio basati sulle tendenze, basati sui fattori di aziendali o su una combinazione di entrambi i metodi di previsione. I budget devono essere calcolati, realistici e basati sulle previsioni.
- Configura avvisi e notifiche: utilizza rilevamento delle anomalie dei costi e avvisi di Budget AWS per ricevere avvisi e notifiche.
- Esegui revisioni periodiche con le principali parti interessate: ad esempio, è necessario allinearsi ai cambiamenti nella direzione dell'azienda e nell'utilizzo con le parti interessate dell'IT, della finanza, dei team della piattaforma e di altre aree dell'azienda.

Risorse

Documenti correlati:

- AWS Cost Explorer
- AWS Cost and Usage Report
- Forecasting with Cost Explorer
- Previsione rapida di Quick
- Budget AWS

Video correlati:

- How can I use Budget AWS to track my spending and usage
- AWS Cost Optimization Series: Budget AWS

Esempi correlati:

- Understand and build driver-based forecasting
- How to establish and drive a forecasting culture
- How to improve your cloud cost forecasting
- Using the right tools for your cloud cost forecasting

COST01-BP04 Implementazione della consapevolezza dei costi nei processi dell'organizzazione

Implementa la consapevolezza dei costi e crea trasparenza e funzionalità di controllo in processi nuovi o esistenti che influiscono sull'utilizzo e sfrutta i processi esistenti per favorire la consapevolezza dei costi. Implementa la consapevolezza dei costi nella formazione dei dipendenti.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

La consapevolezza dei costi deve essere implementata nei processi organizzativi nuovi ed esistenti. Si tratta di un prerequisito fondamentale per altre best practice. È consigliabile riutilizzare e modificare i processi esistenti, laddove possibile, riducendo al minimo l'impatto sull'agilità e sulla velocità. Comunica i costi del cloud ai team tecnologici e ai responsabili dei processi decisionali nei team aziendali e finanziari per accrescere la consapevolezza dei costi e definisci indicatori chiave delle prestazioni (KPI) per l'efficienza da comunicare alle parti interessate nelle varie aree finanziarie e aziendali. Le seguenti raccomandazioni aiuteranno a implementare la consapevolezza dei costi nel carico di lavoro:

- Verifica che la gestione delle modifiche includa una misurazione dei costi per quantificare l'impatto finanziario delle modifiche. Questo aiuta a risolvere in modo proattivo le problematiche relative ai costi nonché a evidenziare i risparmi ottenuti.
- Verifica che l'ottimizzazione dei costi sia un componente fondamentale delle tue capacità operative. Ad esempio, puoi sfruttare gli attuali processi di gestione degli incidenti per analizzare e identificare la causa principale di anomalie di costi e utilizzo o delle eccedenze di costo.

- Accelera la riduzione dei costi e la realizzazione del valore aggiunto attraverso l'automazione o l'utilizzo di strumenti. Quando valuti i costi dell'implementazione, includi nella valutazione un componente ROI per giustificare l'investimento di tempo o denaro.
- Assegna i costi del cloud mediante l'implementazione delle policy di showback/chargeback per la spesa cloud, compresa la spesa per opzioni di acquisto basate su impegno, servizi condivisi e acquisti su marketplace, a supporto di un consumo del cloud maggiormente consapevole dei costi.
- Estendi i programmi di formazione e sviluppo esistenti per includere la formazione sulla consapevolezza dei costi in tutta l'organizzazione, comprese attività di formazione continua e certificazione. In questo modo, creerai un'organizzazione in grado di gestire in modo autonomo i costi e l'utilizzo.
- Sfrutta i vantaggi degli strumenti nativi AWS gratuiti, come AWS Cost Anomaly Detection, Budget AWS e i report di Budget AWS.

Se le organizzazioni adottano in modo costante le pratiche di Cloud Financial Management (CFM), i comportamenti in questione si radicano nel modo di lavorare e nel processo decisionale. Ne risulterà una cultura basata su una maggiore consapevolezza dei costi, condivisa dagli sviluppatori che creano nuove applicazioni per il cloud e dai responsabili dell'area finanziaria che analizzano il ROI per questi nuovi investimenti a livello di cloud.

Passaggi dell'implementazione

- Identificazione dei processi organizzativi pertinenti: ciascuna unità organizzativa esamina i propri processi e identifica quelli che influiscono su costi e utilizzo. Tutti i processi che determinano la creazione o la cessazione di una risorsa devono essere inclusi nella revisione. Individua i processi che possono supportare la consapevolezza dei costi nella tua azienda, ad esempio la gestione degli incidenti e la formazione.
- Definizione di una cultura consapevole dei costi autosufficiente: assicurati che tutte le parti interessate pertinenti siano concordi sulla causa della modifica e sull'impatto come costo in modo che abbiano la piena consapevolezza del costo del cloud. Ciò consentirà all'organizzazione di definire una cultura consapevole dei costi autosufficiente finalizzata all'innovazione.
- Aggiornamento dei processi con la consapevolezza dei costi: la modifica dei processi avviene per renderli consapevoli dei costi. Il processo potrebbe richiedere ulteriori controlli preliminari, ad esempio la valutazione dell'impatto dei costi, oppure controlli successivi che attestino il verificarsi dei cambiamenti previsti in termini di costi e utilizzo. I processi di supporto come la formazione e la gestione degli incidenti possono essere estesi per includere elementi relativi a costi e utilizzo.

Per ottenere assistenza, contatta gli CFM mediante il team del tuo account oppure esplora le risorse e i documenti correlati elencati di seguito.

Risorse

Documenti correlati:

- Gestione finanziaria del cloud con AWS

Esempi correlati:

- Strategy for Efficient Cloud Cost Management
- Cost Control Blog Series #3: How to Handle Cost Shock
- A Beginner's Guide to AWS Cost Management

COST01-BP05 Invio di report e notifiche sull'ottimizzazione dei costi

Imposta i budget per il cloud e configura i meccanismi per rilevare anomalie nell'utilizzo. Configura gli strumenti correlati per ricevere avvisi su costi e utilizzo rispetto a obiettivi predefiniti e ricevi notifiche quando l'utilizzo supera tali obiettivi. Organizza riunioni regolari per analizzare l'economicità dei tuoi carichi di lavoro e promuovere la consapevolezza dei costi.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

È necessario inviare report regolari sull'ottimizzazione dei costi e sull'utilizzo all'interno dell'organizzazione. Puoi implementare sessioni dedicate per discutere le prestazioni in termini di costi o includere l'ottimizzazione dei costi nei regolari cicli di rendicontazione operativi per i tuoi carichi di lavoro. Utilizza servizi e strumenti per monitorare regolarmente le prestazioni in termini di costi e implementare opportunità di risparmio sui costi.

Visualizza i costi e l'utilizzo con più filtri e granularità utilizzando AWS Cost Explorer, che fornisce pannelli di controllo e report come i costi per servizio o per account, i costi giornalieri o i costi del marketplace. Monitora l'avanzamento di costi e utilizzo rispetto ai budget configurati attraverso i report di Budget AWS

Utilizza Budget AWS per impostare budget personalizzati in modo da tenere traccia di costi e utilizzo e rispondere rapidamente agli avvisi ricevuti tramite e-mail o alle notifiche di Amazon Simple Notification Service (Amazon SNS) in caso di superamento della soglia. Imposta il periodo di budget

preferito su giornaliero, mensile, trimestrale o annuale, quindi crea limiti di budget specifici così da ricevere in modo costante informazioni sull'andamento di costi e utilizzi effettivi o previsti rispetto alla soglia del tuo budget. Puoi anche configurare avvisi e azioni da eseguire automaticamente o in base a un processo di approvazione a fronte di tali avvisi quando viene superato l'obiettivo del budget.

Implementa notifiche su costi e utilizzo per reagire in modo rapido a variazioni di costi e utilizzi imprevisti. AWS Cost Anomaly Detection consente di ridurre gli imprevisti in termini relativi ai costi e migliorare il controllo senza rallentare l'innovazione. AWS Cost Anomaly Detection individua le spese anomale e le cause principali, in modo da ridurre il rischio di imprevisti nella fatturazione. Grazie a tre semplici passaggi, è possibile creare una funzione di controllo contestualizzata personalizzata e ricevere avvisi quando viene rilevata una spesa anomala.

Puoi anche utilizzare Quick con dati AWS Cost and Usage Report (CUR) per fornire report altamente personalizzati con dati più granulari. Con Quick Suite puoi pianificare report e ricevere e-mail periodiche con report sui costi in relazione a costi e utilizzo a livello cronologico oppure opportunità di risparmio sui costi. Scopri la nostra Cost Intelligence Dashboard (CID) basata su Quick, per una visibilità avanzata.

Usa AWS Trusted Advisor, che mette a disposizione linee guida per verificare se le risorse allocate sono conformi alle best practice AWS in relazione all'ottimizzazione dei costi.

Controlla le tue raccomandazioni Savings Plans tramite grafici visivi confrontandoli con i costi e l'utilizzo granulari. I grafici orari mostrano la spesa on demand insieme all'impegno verso i Savings Plans raccomandati, fornendo informazioni sui risparmi stimati, sulla copertura dei Savings Plans e sull'utilizzo dei Savings Plans. Questo aiuta le organizzazioni a capire in che modo i loro Savings Plans si applicano a ogni ora di spesa senza dover investire tempo e risorse nella creazione di modelli per analizzare la spesa stessa.

Crea periodicamente report contenenti informazioni di primo piano relative a Savings Plans, istanze riservate e suggerimenti per il ridimensionamento corretto di Amazon EC2 provenienti da AWS Cost Explorer per favorire la riduzione dei costi associati a carichi di lavoro con stato stazionario e a risorse inattive e sottoutilizzate. Individua e ammortizza la spesa associata all'utilizzo non ottimale del cloud relativamente alle risorse implementate. Con utilizzo non ottimale del cloud si intende la creazione di risorse dimensioni errate oppure la presenza di modelli di utilizzo del cloud diversi da quanto previsto. Segui le best practice di AWS per ridurre gli sprechi o chiedi al team del tuo account o al tuo partner di aiutarti a ottimizzare e risparmiare sui tuoi costi del cloud.

Genera regolarmente report per migliorare le opzioni di acquisto delle risorse al fine di ridurre il costo unitario dei carichi di lavoro. Le opzioni di acquisto quali, ad esempio, Savings Plans, istanze

riservate o istanze spot di Amazon EC2, offrono il massimo risparmio sui costi per carichi di lavoro con tolleranza ai guasti, consentendo alle parti coinvolte (proprietari di aziende, team finanziari e tecnologici) di venire coinvolti nelle discussioni di merito.

Condividi i report contenenti opportunità o annunci di nuovi rilasci a supporto della riduzione del costo totale di proprietà (TCO) del cloud. Adotta nuovi servizi, regioni, funzionalità, soluzioni o nuovi modi per migliorare ulteriormente la riduzione dei costi.

Passaggi dell'implementazione

- Configura Budget AWS: configura Budget AWS su tutti gli account per il tuo carico di lavoro. Imposta un budget per la spesa complessiva dell'account e un budget per il carico di lavoro utilizzando i tag.
  - Well-Architected Labs: utilizzo di costi e governance
- Crea report sull'ottimizzazione dei costi: configura un ciclo regolare per discutere e analizzare l'efficienza del carico di lavoro. Utilizzando i parametri stabiliti, segnala i parametri raggiunti e il costo sostenuto per ottenerli. Identifica e correggi eventuali tendenze negative e individua tendenze positive che puoi favorire in tutta l'organizzazione. La rendicontazione dovrebbe coinvolgere i rappresentanti dei team e dei responsabili delle applicazioni, dei responsabili finanziari e dei principali responsabili delle decisioni in merito alla spesa per il cloud.

Risorse

Documenti correlati:

- AWS Cost Explorer
- AWS Trusted Advisor
- Budget AWS
- AWS Cost and Usage Report
- Budget AWS Best practice di
- Amazon S3 Analytics

Esempi correlati:

- Key ways to start optimizing your AWS cloud costs

COST01-BP06 Monitoraggio proattivo dei costi

Implementa strumenti e pannelli di controllo per monitorare i costi in modo proattivo per il carico di lavoro. Rivedi regolarmente i costi utilizzando strumenti configurati o pronti all'uso e non limitarti a guardare solo i costi e le categorie quando ricevi le notifiche. Il monitoraggio e l'analisi dei costi aiutano in modo proattivo a individuare i trend positivi e a promuoverli nell'organizzazione.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Si consiglia di monitorare i costi e l'utilizzo all'interno dell'organizzazione in modo proattivo, e non solo in caso di eccezioni o anomalie. I pannelli di controllo con un'elevata visibilità in tutto l'ufficio o l'ambiente di lavoro garantiscono che le persone chiave abbiano accesso alle informazioni di cui hanno bisogno e dimostrano l'attenzione che l'organizzazione presta all'ottimizzazione dei costi. I pannelli di controllo visibili consentono di promuovere attivamente i risultati positivi e di implementarli in tutta l'organizzazione.

Crea una routine quotidiana o frequente volta all'utilizzo di AWS Cost Explorer o qualsiasi altro pannello di controllo come Amazon Quick per scoprire i costi ed effettuare analisi in modo proattivo. Analizza l'utilizzo e i costi dei servizi AWS a livello di account AWS, carico di lavoro o servizio AWS specifico in gruppo o mediante filtri e verifica che siano in linea con quanto previsto. Utilizza tag e granularità a livello orario o di risorsa per filtrare e individuare i costi ricorrenti relativi alle risorse di maggiore utilizzo. Puoi anche creare report personalizzati con Cost Intelligence Dashboard, una soluzione Amazon Quick sviluppata da AWS Solutions Architect, e confrontare i tuoi budget con i costi e l'utilizzo effettivi.

Passaggi dell'implementazione

- Crea report sull'ottimizzazione dei costi: configura un ciclo regolare per discutere e analizzare l'efficienza del carico di lavoro. Utilizzando i parametri stabiliti, segnala i parametri raggiunti e il costo sostenuto per ottenerli. Identifica e correggi eventuali tendenze negative e identifica le tendenze positive che puoi favorire in tutta l'organizzazione. L'elaborazione dei report deve coinvolgere i rappresentanti dei team applicativi e dei proprietari, dei team finanziari e di gestione.
- Creazione e attivazione di Budget AWS con granularità giornaliera relativa a costi e utilizzo per adottare misure tempestive volte a impedire potenziali superamenti dei costi: Budget AWS ti permette di configurare notifiche di avviso. In questo modo, riceverai sempre informazioni se uno dei tuoi tipi di budget supera le soglie preconfigurate. Il modo migliore per utilizzare Budget AWS

è configurare i costi e l'utilizzo previsti come limite in modo tale che qualsiasi superamento del budget possa essere considerato un superamento del limite di spesa.

- Creazione di AWS Cost Anomaly Detection per il monitoraggio dei costi: AWS Cost Anomaly Detection utilizza la tecnologia avanzata di machine learning per individuare le spese anomale e le cause principali in modo da garantire un intervento tempestivo. Ti consente di configurare funzionalità di monitoraggio dei costi che definiscono i segmenti di spesa da valutare, ad esempio singoli servizi AWS, account membro, tag di allocazione dei costi e categorie di costo, nonché di impostare quando, dove e come riceverai le notifiche di avviso. Per ciascuna funzionalità di monitoraggio, puoi associare più sottoscrizioni agli avvisi per proprietari di azienda e team tecnologici, inclusi un nome, una soglia relativa all'impatto dei costi e la frequenza di avviso (avvisi singoli, riepilogo giornaliero, riepilogo settimanale) per ciascuna sottoscrizione.
- Utilizzo di AWS Cost Explorer o integrazione dei dati AWS Cost and Usage Report (CUR) con i pannelli di controllo di Amazon Quick per la visualizzazione dei costi della tua organizzazione: AWS Cost Explorer offre un'interfaccia intuitiva per visualizzare, analizzare e gestire costi e utilizzo di AWS nel tempo. Cost Intelligence Dashboard è personalizzabile e accessibile e consente di creare le basi di uno strumento di gestione e ottimizzazione dei costi personalizzato.

Risorse

Documenti correlati:

- Budget AWS
- AWS Cost Explorer
- Daily Cost and Usage Budgets
- AWS Cost Anomaly Detection

Esempi correlati:

- AWS Cost Anomaly Detection Alert with Slack

COST01-BP07 Resta aggiornato up-to-date sulle nuove release di servizio

Consulta regolarmente esperti o AWS partner per valutare quali servizi e funzionalità offrono costi inferiori. AWS Consulta blog e altre fonti di informazione.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

AWS aggiunge costantemente nuove funzionalità in modo da poter sfruttare le tecnologie più recenti per sperimentare e innovare più rapidamente. Potresti essere in grado di implementare nuovi AWS servizi e funzionalità per aumentare l'efficienza dei costi del tuo carico di lavoro. Consulta regolarmente la pagina sulla gestione dei costi AWS, il blog delle novitàAWS, il blog sulla gestione dei costi AWS, e Novità di AWS per informazioni su nuovi servizi e lanci di funzionalità. I post sulle novità forniscono una breve panoramica di tutti gli annunci relativi a AWS servizi, funzionalità e aree geografiche non appena vengono pubblicati.

Passaggi dell'implementazione

- Iscriviti ai blog: vai alle pagine dei AWS blog e iscriviti al blog What's New e ad altri blog pertinenti. Puoi registrarti nella pagina delle preferenze di comunicazione con il tuo indirizzo e-mail.
- Iscriviti alle AWS notizie: consulta regolarmente il AWS News Blog e What's New with AWS per informazioni sulle nuove versioni di servizi e funzionalità. Iscriviti al RSS feed o con la tua email per seguire gli annunci e i comunicati.
- Segui le riduzioni AWS dei prezzi: le riduzioni regolari dei prezzi di tutti i nostri servizi sono state un modo standard per trasferire AWS ai nostri clienti le efficienze economiche ottenute grazie alla nostra scala. Al 20 settembre 2023, AWS ha ridotto i prezzi 134 volte dal 2006. Se hai ancora qualche dubbio in merito a decisioni commerciali da prendere a causa di questioni relative ai prezzi, puoi fare riferimento ai nuovi tariffari, che includono riduzioni dei prezzi e nuove integrazioni dei servizi. Puoi scoprire le precedenti iniziative di riduzione dei prezzi, comprese le istanze Amazon Elastic Compute Cloud EC2 (Amazon), nella categoria riduzione dei prezzi del News Blog. AWS
- AWS eventi e meetup: partecipa al AWS summit locale e a qualsiasi incontro locale con altre organizzazioni della tua zona. Se non puoi partecipare di persona, prova a partecipare agli eventi virtuali per conoscere meglio gli AWS esperti e i casi aziendali di altri clienti.
- Organizza riunioni con il team del tuo account: pianifica una cadenza regolare di incontri con il team del tuo account, organizza riunioni con il team e discuti delle tendenze del settore e dei servizi AWS . Parla con gli account manager, i solutions architect e i team di supporto a te assegnati.

Risorse

Documenti correlati:

- AWS Gestione dei costi

- Cosa c'è di nuovo con AWS
- AWS Blog di notizie

Esempi correlati:

- AmazonEC2: 15 anni di ottimizzazione e risparmio dei costi IT
- AWS News Blog - Riduzione dei prezzi

COST01-BP08 Creazione di una cultura consapevole dei costi

Implementa modifiche o programmi all'interno dell'organizzazione per creare una cultura consapevole dei costi. Si consiglia di iniziare in piccolo, per poi implementare programmi di grandi dimensioni e di vasta portata all'aumentare delle capacità e dell'utilizzo del cloud da parte dell'organizzazione.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Una cultura consapevole dei costi consente di scalare l'ottimizzazione e la gestione finanziaria del cloud (team operativi e finanziari, centro di eccellenza del cloud, operazioni nel cloud e così via) attraverso best practice eseguite in modo organico e decentralizzato all'interno di tutta l'organizzazione. La consapevolezza dei costi crea livelli elevati di capacità all'interno dell'organizzazione con uno sforzo minimo, qualcosa di analogo a un approccio centralizzato e dall'alto verso il basso.

La creazione della consapevolezza dei costi nel cloud computing, soprattutto per quanto riguarda i principali fattori dei costi, consente ai team di avere la piena consapevolezza dei risultati previsti associati a qualsiasi variazione a livello di costi. I team con accesso agli ambienti cloud devono conoscere i modelli dei prezzi e la differenza tra i tradizionali data center on-premises e il cloud computing.

Il principale vantaggio di una cultura consapevole dei costi è che i team tecnologici ottimizzano i costi in modo proattivo e continuativo (ad esempio, i costi vengono considerati un requisito non funzionale durante la definizione dell'architettura dei nuovi carichi di lavoro oppure quando vengono apportate modifiche ai carichi di lavoro esistenti) anziché eseguire ottimizzazioni reattive dei costi, in caso di necessità.

Piccoli cambiamenti nella cultura possono avere un grande impatto sull'efficienza dei carichi di lavoro attuali e futuri. Esempi di questo tipo includono:

- Avere visibilità e consapevolezza consente ai team tecnici di progettazione di controllare il loro operato e di capire il tipo di impatto che la loro attività ha in termini di costi.
- Gamificare costi e utilizzo in tutta l'organizzazione. Questa operazione può essere eseguita tramite un pannello di controllo visibile pubblicamente o un report che confronta i costi e l'utilizzo normalizzati tra i team (ad esempio, i costi per carico di lavoro e i costi per transazione).
- Premiare l'efficienza dei costi. Ricompensa pubblicamente o privatamente i risultati di ottimizzazione dei costi volontari o non sollecitati e impara dagli errori per evitare di ripeterli in futuro.
- Crea requisiti organizzativi dall'alto verso il basso affinché i carichi di lavoro siano eseguiti nel rispetto dei budget predefiniti.
- Esegui una verifica continua dei requisiti aziendali relativi alle modifiche e dell'impatto dei costi delle modifiche richieste sull'infrastruttura dell'architettura o sulla configurazione del carico di lavoro per avere la certezza di pagare solo quanto è necessario.
- Verifica che il responsabile delle modifiche sia consapevole delle modifiche previste con un impatto sui costi, che a loro volta devono essere confermate dalle parti coinvolte al fine di ottenere risultati aziendali in modo economicamente conveniente.

Passaggi dell'implementazione

- Comunica i costi del cloud ai team tecnologici: per favorire la consapevolezza dei costi e definire indicatori KPI relativi all'efficienza per le parti coinvolte nelle aree finanziarie e aziendali.
- Comunica le modifiche pianificate alle parti interessate o ai membri dei team: crea una voce nel programma per discutere le modifiche pianificate e l'impatto costi/benefici a livello di carico di lavoro durante le riunioni settimanali.
- Organizza riunioni con il team del tuo account: pianifica una cadenza regolare di incontri con il team del tuo account e discuti delle tendenze del settore e dei servizi AWS. Parla con account manager, architect e team di supporto a te assegnati.
- Condividi le storie di successo: condividi le storie di successo relative alla riduzione dei costi per qualsiasi carico di lavoro, Account AWS o organizzazione per creare un atteggiamento favorevole e incoraggiare la consapevolezza a questo proposito.
- Formazione: assicurati che i team tecnici o i membri dei vari team abbiano ricevuto una formazione adeguata in merito alla consapevolezza dei costi delle risorse nel Cloud AWS.
- Eventi e incontri AWS: partecipa ai summit AWS locali e a qualsiasi incontro locale con altre organizzazioni della tua area.

- Iscriviti ai blog: vai alle pagine dei blog AWS e iscriviti al blog delle novità e ad altri blog pertinenti per seguire nuove versioni, implementazioni, esempi e modifiche condivise da AWS.

Risorse

Documenti correlati:

- Blog AWS
- Gestione dei costi AWS
- Blog delle novità AWS

Esempi correlati:

- Gestione finanziaria del cloud con AWS

COST01-BP09 Quantifica il valore aziendale grazie all'ottimizzazione dei costi

La quantificazione del valore aggiunto realizzato tramite l'ottimizzazione dei costi consente di comprendere l'intero set di vantaggi per la tua organizzazione. Poiché l'ottimizzazione dei costi è un investimento necessario, la quantificazione del valore aggiunto consente di spiegare il ritorno sull'investimento alle parti interessate. La quantificazione del valore aggiunto può aiutarti a ottenere maggiori consensi dalle parti interessate sugli investimenti futuri in materia di ottimizzazione dei costi, e fornisce un framework per misurare i risultati delle attività di ottimizzazione dei costi della tua organizzazione.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Quantificare il valore aziendale significa misurare i vantaggi che le aziende ottengono dalle azioni e dalle decisioni che prendono. Il valore aziendale può essere tangibile (riduzione delle spese o aumento dei profitti) o intangibile (migliore reputazione del marchio o maggiore soddisfazione del cliente).

Quantificare il valore aziendale derivante dall'ottimizzazione dei costi significa determinare il valore o i vantaggi ottenuti dall'impegno dedicato a rendere più efficiente la spesa. Ad esempio, se un'azienda spende 100.000 dollari per implementare un carico di lavoro AWS e successivamente lo ottimizza, il nuovo costo diventa di soli 80.000 dollari senza sacrificare la qualità o l'output. In questo scenario, il

valore aziendale quantificato derivante dall'ottimizzazione dei costi è un risparmio di 20.000 dollari. Ma oltre ai semplici risparmi, l'azienda potrebbe anche quantificare il valore in termini di tempi di consegna più rapidi, maggiore soddisfazione dei clienti o altre metriche derivanti dall'impegno nell'ambito dell'ottimizzazione dei costi. Le parti interessate devono prendere decisioni in merito al potenziale valore dell'ottimizzazione dei costi, al costo dell'ottimizzazione del carico di lavoro e al valore del ritorno sugli investimenti.

Oltre a rendicontare i risparmi derivanti dall'ottimizzazione dei costi, è consigliabile quantificare il valore aggiunto fornito. I vantaggi dell'ottimizzazione dei costi sono in genere quantificati in termini di costi inferiori per ottenere un risultato aziendale. Ad esempio, puoi quantificare Amazon Elastic Compute Cloud(AmazonEC2) i risparmi sui costi acquistando Savings Plans, che riducono i costi e mantengono i livelli di output del carico di lavoro. Puoi quantificare le riduzioni dei costi di AWS spesa quando le istanze EC2 Amazon inattive vengono rimosse o i volumi Amazon Elastic Block Store (EBSAmazon) non collegati vengono eliminati.

I vantaggi derivanti dall'ottimizzazione dei costi, tuttavia, vanno oltre la riduzione o l'eliminazione dei costi. Prendi in considerazione l'acquisizione di dati aggiuntivi per misurare i miglioramenti dell'efficienza e il valore aggiunto.

Passaggi dell'implementazione

- Valuta i vantaggi aziendali: questo è il processo di analisi e regolazione dei Cloud AWS costi in modo da massimizzare il beneficio ricevuto da ogni dollaro speso. Invece di concentrarti sulla riduzione dei costi senza considerare il valore aziendale, nell'ambito dell'ottimizzazione dei costi valuta i vantaggi aziendali e il ritorno sugli investimenti, che potrebbero aumentare il valore del denaro speso. Si tratta di spendere con saggezza e di fare investimenti e spese nelle aree che producono i migliori rendimenti.
- Analisi AWS dei costi di previsione: le previsioni aiutano gli stakeholder finanziari a stabilire le aspettative con gli altri stakeholder interni ed esterni dell'organizzazione e possono migliorare la prevedibilità finanziaria dell'organizzazione. AWS Cost Explorerpuò essere utilizzato per eseguire previsioni relative ai costi e all'utilizzo.

Risorse

Documenti correlati:

- Cloud AWS Economia
- AWS Blog

- AWS Gestione dei costi
- AWS Blog di notizie
- Whitepaper sul pilastro dell'affidabilità Well-Architected
- AWS Cost Explorer

Video correlati:

- Sblocca il valore aziendale con Windows on AWS

Esempi correlati:

- Measuring and Maximizing the Business Value of Customer 360
- The Business Value of Adopting Amazon Web Services Managed Databases
- The Business Value of Amazon Web Services for Independent Software Vendors
- Business Value of Cloud Modernization
- The Business Value of Migration to Amazon Web Services

Comprensione delle spese e dell'utilizzo

Questions

- COST 2. In che modo gestisci l'utilizzo?
- COST 3. In che modo monitori i costi e l'utilizzo?
- COST 4. In che modo disattivi le risorse?

COST 2. In che modo gestisci l'utilizzo?

Stabilisci policy e meccanismi per verificare che i costi sostenuti mentre raggiungi gli obiettivi siano adeguati. Utilizzando un approccio di controllo e bilanciamento reciproco, è possibile innovare senza spendere troppo.

Best practice

- COST02-BP01 Sviluppo di policy basate sui requisiti dell'organizzazione
- COST02-BP02 Implementazione di obiettivi e target
- COST02-BP03 Implementazione di una struttura di account

- COST02-BP04 Implementazione di gruppi e ruoli
- COST02-BP05 Implementazione dei controlli di costo
- COST02-BP06 Tieni traccia del ciclo di vita del progetto

COST02-BP01 Sviluppo di policy basate sui requisiti dell'organizzazione

Sviluppa policy che definiscano il modo in cui le risorse vengono gestite dalla tua organizzazione e controllale periodicamente. Le policy devono coprire gli aspetti dei costi relativi alle risorse e ai carichi di lavoro, comprese la creazione, la modifica e la disattivazione nel ciclo di vita delle risorse.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Comprendere i costi e i fattori chiave della tua organizzazione è fondamentale per gestire i costi e l'utilizzo in modo efficiente e per identificare le opportunità di riduzione dei costi. In genere, le organizzazioni gestiscono molteplici carichi di lavoro eseguiti da più team. Questi team possono trovarsi in diverse unità dell'organizzazione, ciascuna con un proprio flusso di ricavi. La capacità di attribuire i costi delle risorse ai singoli proprietari del carico di lavoro, del prodotto o dell'organizzazione incoraggia un comportamento di utilizzo efficiente e contribuisce a ridurre gli sprechi. Un monitoraggio accurato dei costi e dell'utilizzo consente di comprendere quanto sia ottimizzato un carico di lavoro e quanto siano redditizi i prodotti e le unità organizzative. Questa conoscenza consente di prendere decisioni più informate su dove allocare le risorse all'interno dell'organizzazione. La consapevolezza dell'utilizzo a tutti i livelli dell'organizzazione è fondamentale per promuovere il cambiamento, poiché la modifica dell'utilizzo determina variazioni dei costi. Prova ad adottare una strategia versatile per acquisire consapevolezza delle tue spese.

Il primo passo per attuare la governance consiste nell'utilizzare i requisiti della tua organizzazione per sviluppare policy per l'utilizzo del cloud. Queste policy definiscono il modo in cui l'organizzazione utilizza il cloud e il modo in cui le risorse vengono gestite. Le policy devono coprire tutti gli aspetti dei costi relativi alle risorse e ai carichi di lavoro correlati a costi o utilizzo, compresa la creazione, la modifica e la disattivazione durante il ciclo di vita di una risorsa. Verifica che policy e procedure vengano eseguite e implementate per qualsiasi modifica apportata in un ambiente cloud. Durante gli incontri per la gestione delle modifiche IT, poni domande relative all'impatto sui costi delle modifiche pianificate (se implicano un aumento o una riduzione), alla giustificazione aziendale e ai risultati attesi.

Le policy devono essere semplici, in modo che siano facilmente comprensibili e possano essere implementate in modo efficace in tutta l'organizzazione. Le policy devono anche essere facili da

seguire e interpretare (in modo da essere utilizzate) e specifiche (senza interpretazioni errate tra i team). Inoltre, devono essere ispezionate periodicamente (come i nostri meccanismi) e aggiornate man mano che le condizioni o le priorità aziendali dei clienti cambiano, il che renderebbe la policy obsoleta.

Inizia con policy ampie e di alto livello, ad esempio in quale regione geografica è consentito l'utilizzo o l'ora del giorno in cui le risorse devono essere in esecuzione. Affina gradualmente le policy per le varie unità organizzative e i diversi carichi di lavoro. Le policy comuni includono i servizi e le funzionalità che possono essere utilizzati (ad esempio, archiviazione dalle prestazioni inferiori negli ambienti di test e sviluppo), i tipi di risorse che possono essere utilizzati dai diversi gruppi (ad esempio, le dimensioni massime di una risorsa in un account di sviluppo possono essere impostate su medie) e per quanto tempo queste risorse saranno in uso (temporaneamente, a breve termine o per un periodo di tempo specifico).

Esempio di policy

Di seguito è riportato un esempio di policy che puoi esaminare per creare le tue policy di governance del cloud, basate sull'ottimizzazione dei costi. Assicurati di adattare la policy ai requisiti della tua organizzazione e alle richieste delle parti interessate.

- Nome della policy: definisci un nome chiaro per la policy, ad esempio Ottimizzazione delle risorse e Policy di riduzione dei costi.
- Scopo: spiega perché questa policy dovrebbe essere utilizzata e qual è il risultato previsto. L'obiettivo di questa policy è verificare che sia richiesto un costo minimo per implementare ed eseguire il carico di lavoro desiderato per soddisfare i requisiti aziendali.
- Ambito di applicazione: definisci chiaramente chi deve utilizzare questa policy e quando deve essere utilizzata, ad esempio Team DevOps X per utilizzare questa policy per i clienti nella zona di disponibilità Stati Uniti-Est per l'ambiente X (di produzione o non di produzione).

Dichiarazione delle policy

1. Seleziona us-east-1 o più regioni Stati Uniti-Est in base all'ambiente del carico di lavoro e ai requisiti aziendali (sviluppo, test di accettazione da parte degli utenti, preproduzione o produzione).
2. Pianifica l'esecuzione delle istanze Amazon EC2 e Amazon RDS tra le sei del mattino e le otto di sera (Ora solare orientale [EST]).
3. Arresta tutte le istanze Amazon EC2 inutilizzate dopo otto ore e le istanze Amazon RDS inutilizzate dopo 24 ore di inattività.

4. Termina tutte le istanze Amazon EC2 inutilizzate dopo 24 ore di inattività in ambienti non di produzione. Ricorda al proprietario dell'istanza Amazon EC2 (in base ai tag) di esaminare le istanze Amazon EC2 arrestate in produzione e informalo che le istanze Amazon EC2 verranno terminate entro 72 ore se non vengono utilizzate.
5. Usa la famiglia e le dimensioni delle istanze generiche come m5.large, quindi ridimensiona l'istanza in base all'utilizzo della CPU e della memoria mediante AWS Compute Optimizer.
6. Assegna la priorità utilizzando il dimensionamento automatico per regolare dinamicamente il numero di istanze in esecuzione in base al traffico.
7. Usa le istanze spot per carichi di lavoro non critici.
8. Esamina i requisiti di capacità per impegnare piani di risparmio o istanze riservate per carichi di lavoro prevedibili e informa il team della gestione finanziaria del cloud.
9. Utilizza le policy Amazon S3 del ciclo di vita per spostare i dati a cui si accede di rado su livelli di archiviazione più economici. Se non è stata definita alcuna policy di conservazione, utilizza il Piano intelligente Amazon S3 per spostare automaticamente gli oggetti nel livello archiviato.
10. Monitora l'utilizzo delle risorse e imposta allarmi per attivare eventi di dimensionamento utilizzando Amazon CloudWatch.
11. Per ogni Account AWS, utilizza Budget AWS per impostare i budget di costo e utilizzo per il tuo account in base al centro di costo e alle business unit.
12. L'utilizzo di Budget AWS per impostare i budget di costi e utilizzo del tuo account può aiutarti a tenere sotto controllo le spese ed evitare fatture impreviste, consentendoti di controllare meglio i costi.

Procedura: fornisci procedure dettagliate per l'attuazione di questa policy o fai riferimento ad altri documenti che descrivono come implementare ciascuna dichiarazione della policy. Questa sezione dovrebbe fornire istruzioni dettagliate per l'adempimento dei requisiti della policy.

Per implementare questa policy, puoi utilizzare vari strumenti o regole AWS Config di terze parti per verificare la conformità alla dichiarazione e attivare azioni correttive automatiche utilizzando le funzioni AWS Lambda. Puoi anche usare AWS Organizations per applicare la policy. Inoltre, dovresti controllare regolarmente l'utilizzo delle risorse e modificare la policy, se necessario, per verificare che continui a soddisfare le esigenze aziendali.

Passaggi dell'implementazione

- Incontra le parti interessate: per sviluppare le policy, chiedi alle parti interessate (ufficio aziendale per il cloud, ingegneri o responsabili delle decisioni funzionali per l'applicazione delle policy)

all'interno della tua organizzazione di specificare i loro requisiti e documentarli. Segui un approccio iterativo iniziando in modo generale e perfezionando continuamente le unità più piccole in ogni fase. I membri del team includono quelli con interesse diretto nel carico di lavoro, ad esempio unità organizzative o proprietari di applicazioni, nonché gruppi di supporto, come i team di sicurezza e i team finanziari.

- Ottieni conferma: verifica che i team siano d'accordo sulle policy a cui possono accedere e che possono distribuire sull'Cloud AWS. Verifica che rispettino le policy della tua organizzazione e conferma che le creazioni di risorse siano in linea con le policy e le procedure concordate.
- Organizza sessioni di formazione per l'onboarding: chiedi ai nuovi membri dell'organizzazione di partecipare a corsi di formazione di onboarding per sviluppare una consapevolezza sui costi e sui requisiti aziendali. Potrebbero adottare policy diverse legate all'esperienza precedente o non rifletterci affatto.
- Definisci le posizioni del tuo carico di lavoro: definisci dove opera il carico di lavoro, incluso il Paese e l'area all'interno del Paese. Queste informazioni vengono utilizzate per la mappatura su Regioni AWS e sulle zone di disponibilità.
- Definisci e raggruppa servizi e risorse: definisci i servizi necessari per il carico di lavoro. Per ogni servizio, specifica i tipi, la dimensione e il numero di risorse richieste. Definisci i gruppi per le risorse in base alla funzione, ad esempio i server di applicazioni o lo storage di database. Le risorse possono appartenere a più gruppi.
- Definisci e raggruppa gli utenti per funzione: definisci gli utenti che interagiscono con il carico di lavoro, concentrandoti su ciò che fanno e su come utilizzano il carico di lavoro, non su chi sono o sulla loro posizione nell'organizzazione. Raggruppa utenti o funzioni simili. Puoi utilizzare le policy gestite da AWS come guida di riferimento.
- Definisci le operazioni: utilizzando le posizioni, le risorse e gli utenti identificati in precedenza, definisci le azioni richieste da ciascuno di essi per ottenere i risultati del carico di lavoro durante il ciclo di vita (sviluppo, funzionamento e disattivazione). Identifica le operazioni in base ai gruppi, non ai singoli elementi nei gruppi, in ogni posizione. Inizia in generale con lettura o scrittura, quindi perfeziona le azioni specifiche per ciascun servizio.
- Definisci il periodo di revisione: carichi di lavoro e requisiti organizzativi possono subire modifiche nel tempo. Definisci la pianificazione della revisione del carico di lavoro per assicurarti che sia allineata alle priorità organizzative.
- Documenta le policy: verifica che le policy definite siano accessibili secondo le esigenze dell'organizzazione. Queste policy vengono utilizzate per implementare, mantenere e controllare l'accesso agli ambienti.

Risorse

Documenti correlati:

- Change Management in the Cloud
- AWS Managed Policies for Job Functions
- Strategia di fatturazione con account multipli di AWS
- Actions, Resources, and Condition Keys for AWS Services
- Gestione e governance su AWS
- Controllo dell'accesso a Regioni AWS utilizzando le policy IAM
- Regioni e zone di disponibilità dell'infrastruttura globale

Video correlati:

- AWS Management and Governance at Scale

COST02-BP02 Implementazione di obiettivi e target

Implementa obiettivi e target di costi e utilizzo per il carico di lavoro. Gli obiettivi forniscono indicazioni alla tua organizzazione sui risultati attesi, mentre i target forniscono risultati misurabili per i tuoi carichi di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Sviluppa obiettivi e target di costi e utilizzo per la tua organizzazione. Per un'organizzazione in crescita su AWS è importante definire e monitorare gli obiettivi ai fini dell'ottimizzazione dei costi. Tali obiettivi o indicatori chiave di prestazione (KPI) possono includere elementi come la percentuale della spesa on demand o l'adozione di determinati servizi ottimizzati come le istanze AWS Graviton o i tipi di volume gp3 EBS. La definizione di obiettivi misurabili e raggiungibili ti aiuta a calcolare i miglioramenti dell'efficienza, un fattore importante per le operazioni aziendali. Gli obiettivi forniscono all'organizzazione linee guida e indicazioni sui risultati previsti.

I target forniscono i risultati specifici e misurabili da raggiungere. In breve, l'obiettivo è la direzione in cui desideri andare, mentre il target è la distanza da percorrere in quella direzione e il momento in cui l'obiettivo deve essere raggiunto, utilizzando la guida SMART, specifica, misurabile, assegnabile,

realistica e tempestiva. Un esempio di obiettivo è che l'utilizzo della piattaforma aumenti in modo significativo, con solo un piccolo incremento (non lineare) dei costi. Un esempio di target è un aumento del 20% dell'utilizzo della piattaforma, con un incremento dei costi inferiore al 5%. Un altro obiettivo comune è che i carichi di lavoro devono essere più efficienti ogni sei mesi. L'obiettivo corrispondente prevede che il costo per metrica aziendale debba diminuire del cinque per cento ogni sei mesi. Usa le metriche giuste e imposta i KPI calcolati per l'organizzazione. Puoi iniziare con i KPI di base e cambiare successivamente in base alle esigenze aziendali.

Un obiettivo per l'ottimizzazione dei costi è l'incremento dell'efficienza del carico di lavoro, ossia la riduzione del costo per ogni risultato aziendale del carico di lavoro nel corso del tempo. Implementa questo obiettivo per tutti i carichi di lavoro e stabilisci un target come l'incremento dell'efficienza del 5% ogni 6-12 mesi. Nel cloud, puoi raggiungere questo target attraverso la definizione della capacità di ottimizzazione dei costi, nonché nuove versioni di servizi e funzionalità.

I target sono i benchmark quantificabili che desideri raggiungere per conseguire i tuoi obiettivi e che confrontano i tuoi risultati effettivi rispetto al target. Stabilisci i benchmark con i KPI per il costo unitario dei servizi di calcolo, come l'adozione di istanze spot, l'adozione di Graviton, i tipi di istanza più recenti e la copertura on demand, dei servizi di archiviazione, come l'adozione di EBS GP3, gli snapshot EBS obsoleti e l'archiviazione standard Amazon S3, oppure dei servizi di database, come i motori open source RDS, l'adozione di Graviton e la copertura on demand. Questi benchmark e KPI possono aiutarti a verificare che i servizi AWS vengano usati nel modo più conveniente.

La tabella seguente fornisce un elenco di metriche standard AWS di riferimento. Ogni organizzazione può avere valori target diversi per questi KPI.

Categoria	KPI (%)	Descrizione
Calcolo	Copertura dell'utilizzo di EC2	Istanze EC2 (in termini di costi o ore) che utilizzano SP+RI +Spot a fronte del totale (in termini di costo o ore) delle istanze EC2
Calcolo	Utilizzo SP/RI di calcolo	Ore SP o RI utilizzate a fronte delle ore SP o RI totali disponibili

Categoria	KPI (%)	Descrizione
Calcolo	Costo EC2-ora	Costo EC2 diviso per il numero di istanze EC2 in esecuzione nell'ora specificata
Calcolo	Costo per vCPU	Costo per vCPU per tutte le istanze
Calcolo	Generazione di istanze più recenti	Percentuale di istanze su Graviton (o altri tipi di istanze di generazione moderna)
Database	Copertura RDS	Istanze RDS (in termini di costi o ore) che utilizzano RI rispetto al totale (in termini di costo o ore) delle istanze RDS
Database	Utilizzo di RDS	Ore SP o RI utilizzate rispetto alle ore SP o RI totali disponibili
Database	Operatività RDS	Costo EC2 diviso per il numero di istanze RDS in esecuzione nell'ora specificata
Database	Generazione di istanze più recenti	Percentuale di istanze su Graviton (o altri tipi di istanze moderne)
Storage	Utilizzo dell'archiviazione	Costo dell'archiviazione ottimizzato (ad esempio Glacier, Deep Archive o Infrequent Access) diviso per il costo totale della stessa

Categoria	KPI (%)	Descrizione
Assegnazione di tag	Risorse prive di tag	Esploratore dei costi: 1. Filtra crediti, sconti, tasse, rimborsi, marketplace e copia l'ultimo costo mensile. 2. Seleziona Mostra solo risorse prive di tag in Cost Explorer 3. Dividi l'importo delle risorse prive di tag per il costo mensile.

Utilizzando questa tabella, stabilisci i valori target o benchmark che devono essere calcolati in base agli obiettivi dell'organizzazione. Per definire KPI accurati e realistici dovrai misurare determinate metriche e comprendere i risultati aziendali per il carico di lavoro. Quando valuti le metriche delle prestazioni di un'organizzazione, tieni in considerazione i vari tipi di metrica che servono a scopi diversi. Queste metriche misurano principalmente le prestazioni e l'efficienza dell'infrastruttura tecnica piuttosto che direttamente l'impatto aziendale complessivo. Ad esempio, possono tenere traccia dei tempi di risposta del server, della latenza della rete o dei tempi di attività del sistema. Queste metriche sono fondamentali per valutare in che misura l'infrastruttura supporta le operazioni tecniche dell'organizzazione. Tuttavia, non forniscono approfondimenti diretti sugli obiettivi aziendali più ampi, come la soddisfazione del cliente, la crescita dei ricavi o la quota di mercato. Per acquisire un quadro completo delle prestazioni aziendali, integra queste metriche dell'efficienza con le metriche aziendali strategiche direttamente correlate ai risultati aziendali.

Ottieni una visibilità quasi in tempo reale sui KPI e sulle relative opportunità di risparmio e monitora lo stato di avanzamento nel tempo. Per iniziare con la definizione e il monitoraggio degli obiettivi KPI, è consigliabile usare il pannello di controllo dei KPI di Cloud Intelligence Dashboards (CID). Sulla base dei dati disponibili nel report di costi e utilizzo (CUR), il pannello di controllo dei KPI fornisce una serie di KPI consigliati per l'ottimizzazione dei costi con la possibilità di definire obiettivi personalizzati e monitorare lo stato di avanzamento nel tempo.

Se disponi di un'altra soluzione per impostare e monitorare gli obiettivi KPI, assicurati che sia adottata da tutte le parti interessate nella gestione finanziaria del cloud della tua organizzazione.

Passaggi dell'implementazione

- Definisci i livelli di utilizzo previsti: parti dai livelli di utilizzo. Coinvolgi i responsabili dell'applicazione, i team di marketing e i team aziendali a livello più ampio per capire quali sono i livelli di utilizzo previsti per il carico di lavoro. Considera in che modo potrà cambiare la domanda dei clienti nel corso del tempo e se ci saranno modifiche dovute a incrementi stagionali o campagne di marketing.
- Definisci risorse e costi del carico di lavoro: una volta definiti i livelli di utilizzo, quantifica le modifiche nelle risorse del carico di lavoro necessarie per soddisfarli. Potresti dover aumentare le dimensioni o il numero di risorse per un componente del carico di lavoro, aumentare il trasferimento dei dati o modificare i componenti del carico di lavoro in un servizio diverso a un livello specifico. Specifica i costi per ciascuno di questi punti e prevedine la variazione in caso di modifica dell'utilizzo.
- Definisci gli obiettivi aziendali: prendendo l'output dalle variazioni previste in termini di utilizzo e costi, combinalo con le modifiche previste nella tecnologia o in qualsiasi programma in esecuzione e sviluppa obiettivi per il carico di lavoro. Gli obiettivi devono riguardare l'utilizzo e il costo, nonché la relazione tra i due. Gli obiettivi devono essere semplici, di alto livello e aiutare le persone a capire cosa si aspetta l'azienda in termini di risultati, come avere la certezza che le risorse non utilizzate rimangano al di sotto di determinati livelli di costo. Non è necessario definire gli obiettivi per ogni tipo di risorsa non utilizzato o definire i costi causati dalle perdite per gli obiettivi e i target. Assicurati che siano disponibili programmi a livello di organizzazione (ad esempio lo sviluppo di competenze come la formazione e l'istruzione), se ci sono variazioni previste dei costi senza variazioni di utilizzo.
- Definisci i target: per ciascuno degli obiettivi definiti, specifica un target misurabile. Se l'obiettivo è aumentare l'efficienza nel carico di lavoro, il target quantifica il miglioramento (generalmente espresso in risultati aziendali per dollaro speso) e il momento in cui sarà efficace. Ad esempio, potresti definire un obiettivo per ridurre al minimo gli sprechi dovuti al provisioning eccessivo. Con questo obiettivo, il target può stabilire che gli sprechi dovuti al provisioning eccessivo di calcolo nel primo livello dei carichi di lavoro di produzione non superino il 10% del costo di calcolo del livello. Inoltre, un secondo target potrebbe stabilire che gli sprechi dovuti al provisioning eccessivo di calcolo nel secondo livello dei carichi di lavoro di produzione non superino il 5% del costo di calcolo del livello.

Risorse

Documenti correlati:

- AWS managed policies for job functions
- Strategia di fatturazione con account multipli di AWS
- Controllo dell'accesso a Regioni AWS utilizzando le policy IAM
- S.M.A.R.T. Goals
- How to track your cost optimization KPIs with the CID KPI Dashboard

Video correlati:

- Well-Architected Labs: obiettivi e target (Livello 100)

Esempi correlati:

- What is a unit metric?
- Selecting a unit metric to support your business
- Unit metrics in practice – lessons learned
- How unit metrics help create alignment between business functions

COST02-BP03 Implementazione di una struttura di account

Implementa una struttura di account che si adatta alla tua organizzazione. In questo modo sarà possibile ripartire e gestire i costi in tutta l'organizzazione.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

AWS Organizations consente di creare molteplici Account AWS che possono aiutare a governare centralmente l'ambiente mentre si procede a scalare i carichi di lavoro su AWS. È possibile modellare la propria gerarchia organizzativa raggruppando gli Account AWS in una struttura di unità organizzative (OU) e creando molteplici Account AWS sotto ogni OU. Per creare una struttura di account, è necessario decidere innanzitutto quale Account AWS sarà l'account di gestione. In seguito, puoi creare nuovi Account AWS o selezionarne di esistenti come account membri in base alla struttura degli account progettata seguendo le best practice per gli account di gestione e le best practice per gli account membri.

È consigliabile disporre sempre di almeno un account di gestione con un account membro collegato, indipendentemente dalle dimensioni dell'organizzazione o dall'utilizzo. Tutte le risorse del carico di

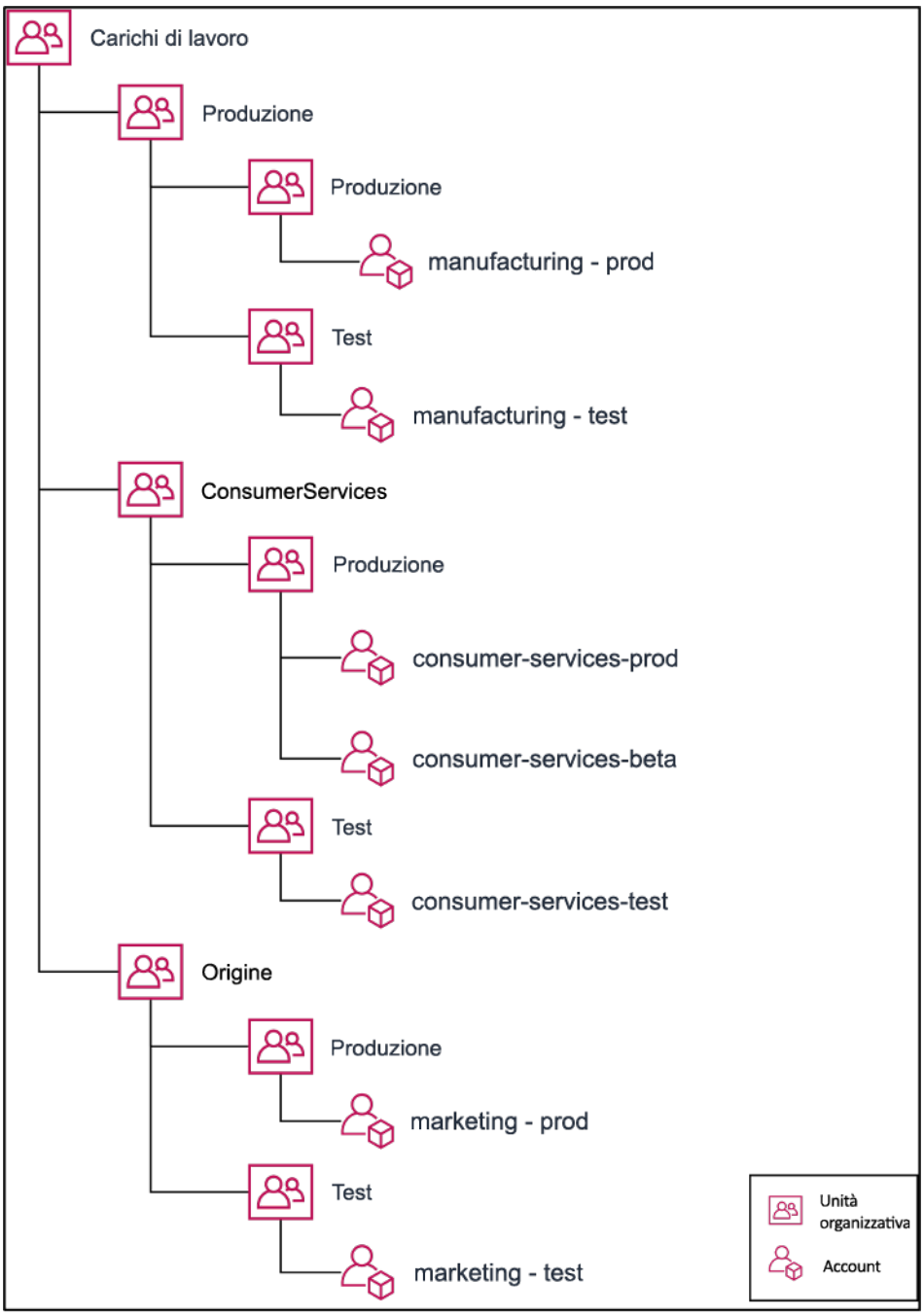
lavoro dovrebbero risiedere solo all'interno degli account membri e nessuna risorsa dovrebbe essere creata all'interno dell'account di gestione. Non esiste una risposta giusta o sbagliata in merito al numero di Account AWS che bisognerebbe creare. Valuta i tuoi modelli operativi e di costo attuali e futuri per assicurarti che la struttura dei tuoi Account AWS rispecchi quella della tua organizzazione. Alcune aziende creano molteplici Account AWS per motivi aziendali, ad esempio:

- È richiesto l'isolamento amministrativo o fiscale e di fatturazione tra unità dell'organizzazione o centri di costo o carichi di lavoro specifici.
- Le restrizioni dei servizi AWS sono impostate in modo che risultino specifiche per determinati carichi di lavoro.
- Esiste un requisito per l'isolamento e la separazione tra carichi di lavoro e risorse.

All'interno di AWS Organizations, la fatturazione consolidata crea il costrutto tra uno o più account membri e l'account di gestione. Gli account membri consentono di isolare e distinguere i costi e l'utilizzo per gruppi. Una pratica comune è quella di avere account membri separati per ciascuna unità dell'organizzazione (come finanza, marketing e vendite), per il ciclo di vita di ciascun ambiente (come sviluppo, test e produzione) o per ciascun carico di lavoro (carico di lavoro a, b e c) e poi aggregare questi account membri tramite la fatturazione consolidata.

La fatturazione consolidata consente di accorpare i pagamenti di più Account AWS membri sotto un unico account di gestione e, al tempo stesso, di fornire comunque visibilità all'attività di ciascun account membro. Il fatto che i costi e l'utilizzo vengono aggregati nell'account di gestione consente di massimizzare gli sconti per volume di servizio e di massimizzare l'utilizzo degli sconti a fronte di impegni (Savings Plans e istanze riservate) per ottenere gli sconti più elevati.

Il diagramma seguente mostra come è possibile utilizzare AWS Organizations con le unità organizzative (OU) per raggruppare più account e come inserire molteplici Account AWS sotto ciascuna OU. Si consiglia di utilizzare le OU per diversi casi d'uso e carichi di lavoro che forniscono modelli per l'organizzazione degli account.



Esempio di raggruppamento di più Account AWS in unità organizzative.

AWS Control Tower può impostare e configurare rapidamente più account AWS, garantendo una governance in linea con i requisiti della tua organizzazione.

Passaggi dell'implementazione

- Definisci i requisiti di separazione: i requisiti di separazione sono una combinazione di più fattori, tra cui sicurezza, affidabilità e costrutti finanziari. Analizza ciascun fattore in ordine e specifica se

il carico di lavoro o l'ambiente del carico di lavoro deve essere separato da altri carichi di lavoro. La sicurezza riguarda il rispetto dei requisiti di accesso e di dati. L'affidabilità riguarda la gestione dei limiti, in modo che gli ambienti e i carichi di lavoro non influiscano gli uni sugli altri. Esamina periodicamente i pilastri della sicurezza e dell'affidabilità del Framework Well-Architected e segui le best practice fornite. I costrutti finanziari creano una rigida separazione finanziaria (centri di costo diversi, proprietà e responsabilità dei carichi di lavoro). Esempi comuni di separazione sono i carichi di lavoro di produzione e test eseguiti in account separati o l'utilizzo di un account separato in modo che i dati di fatturazione possano essere forniti ai singoli settori o reparti aziendali dell'organizzazione o alle terze parti proprietarie dell'account.

- Definisci i requisiti di raggruppamento: i requisiti per il raggruppamento non sostituiscono i requisiti di separazione, ma vengono utilizzati a supporto della gestione. Raggruppa ambienti o carichi di lavoro simili che non richiedono separazione. Un esempio è costituito dal raggruppamento di più ambienti di test o sviluppo associati a uno o più carichi di lavoro.
- Definisci la struttura dell'account: utilizzando queste separazioni e questi raggruppamenti, specifica un account per ciascun gruppo e mantieni i requisiti di separazione. Questi account sono i tuoi account membri o collegati. Raggruppando questi account membri in un unico account di gestione o di pagamento, puoi combinarne l'utilizzo, ottenendo maggiori sconti per i volumi e una singola fattura per tutti gli account. È possibile separare i dati di fatturazione e fornire a ciascun account membro una visualizzazione individuale dei dati di fatturazione. Se un account membro non deve avere i dati di utilizzo o di fatturazione visibili a qualsiasi altro account, oppure se è necessaria una fattura separata da parte di AWS, definisci più account di gestione/di pagamento. In questo caso, ciascun account membro dispone del proprio account di gestione o di pagamento. Le risorse devono sempre essere collocate negli account membri o collegati. Gli account di gestione/di pagamento devono essere utilizzati solo per la gestione.

Risorse

Documenti correlati:

- Utilizzo dei tag per l'allocazione dei costi
- AWS managed policies for job functions
- Strategia di fatturazione con account multipli di AWS
- Controllo dell'accesso a Regioni AWS utilizzando le policy IAM
- AWS Control Tower
- AWS Organizations

- Best practice per account di gestione e account membri
- Organizzazione dell'ambiente AWS che utilizza più account
- Attivazione della condivisione di sconti istanze riservate e Savings Plans
- Fatturazione consolidata
- Fatturazione consolidata

Esempi correlati:

- Divisione della CUR e condivisione dell'accesso

Video correlati:

- Introducing AWS Organizations
- Set Up a Multi-Account AWS Environment that Uses Best Practices for AWS Organizations

Esempi correlati:

- Defining an AWS Multi-Account Strategy for telecommunications companies
- Best practice per l'ottimizzazione dei costi Account AWS
- Best Practices for Organizational Units with AWS Organizations

COST02-BP04 Implementazione di gruppi e ruoli

Implementa gruppi e ruoli che si allineino alle tue policy e controlla chi può creare, modificare o ritirare istanze e risorse in ogni gruppo. Ad esempio, implementa gruppi di sviluppo, test e produzione. Questo si applica ai servizi AWS e a soluzioni di terze parti.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

I ruoli e i gruppi di utenti sono elementi costitutivi fondamentali nella progettazione e implementazione di sistemi sicuri ed efficienti. I ruoli e i gruppi aiutano le organizzazioni a trovare il giusto equilibrio a livello di controllo dei requisiti di flessibilità e produttività, supportando in definitiva gli obiettivi organizzativi e le esigenze degli utenti. Come consigliato nella sezione Gestione delle identità e degli accessi del pilastro della sicurezza del Framework AWS Well-Architected, occorre predisporre una

gestione affidabile di identità e autorizzazioni per fornire l'accesso alle risorse giuste alle persone giuste nelle giuste condizioni. Gli utenti disporranno solo del livello di accesso necessario per completare le proprie attività. Ciò riduce al minimo il rischio associato all'accesso non autorizzato o all'uso improprio.

Dopo avere sviluppato le policy, è possibile creare gruppi logici e ruoli degli utenti all'interno dell'organizzazione. Ciò consente di assegnare le autorizzazioni, controllare l'utilizzo e semplificare l'implementazione di affidabili meccanismi di controllo degli accessi, impedendo l'accesso non autorizzato alle informazioni sensibili. Inizia con i raggruppamenti di persone di alto livello. Generalmente, questi corrispondono alle unità organizzative e ai ruoli lavorativi (ad esempio, amministratore di sistema nel reparto IT, controllore finanziario o business analyst). I gruppi classificano le persone che eseguono attività simili e necessitano di un accesso simile. I ruoli definiscono che cosa un gruppo deve fare. È più facile gestire le autorizzazioni per gruppi e ruoli che per i singoli utenti. I ruoli e i gruppi assegnano le autorizzazioni in modo coerente e sistematico a tutti gli utenti, evitando errori e incongruenze.

Quando il ruolo di un utente cambia, gli amministratori possono modificare l'accesso a livello di ruolo o di gruppo, anziché riconfigurare i singoli account utente. Ad esempio, un amministratore di sistema nel reparto IT deve disporre di un accesso che permetta di creare tutte le risorse, mentre un membro del team di analisi ha la necessità di creare soltanto risorse di analisi.

Passaggi dell'implementazione

- Implementazione dei gruppi: utilizzando i gruppi di utenti definiti nelle policy dell'organizzazione, implementa i gruppi corrispondenti, se necessario. Per le best practice su utenti, gruppi e autenticazione, consulta il pilastro della sicurezza del Framework AWS Well-Architected.
- Implementazione di ruoli e policy: utilizzando le operazioni definite nelle policy dell'organizzazione, crea le policy di accesso e i ruoli necessari. Per le best practice su ruoli e policy, consulta il pilastro della sicurezza del Framework AWS Well-Architected.

Risorse

Documenti correlati:

- AWS managed policies for job functions
- Strategia di fatturazione con account multipli di AWS
- Pilastro della sicurezza del Framework AWS Well-Architected
- AWS Identity and Access Management (IAM)

- Policy AWS Identity and Access Management

Video correlati:

- Why use Identity and Access Management

Esempi correlati:

- Controllo dell'accesso a Regioni AWS utilizzando le policy IAM
- Starting your Cloud Financial Management journey: Cloud cost operations

COST02-BP05 Implementazione dei controlli di costo

Implementa controlli basati sulle policy dell'organizzazione e sui gruppi e sui ruoli definiti. Questi garantiscono che i costi siano sostenuti solo in base ai requisiti dell'organizzazione come, ad esempio, il controllo dell'accesso alle regioni o ai tipi di risorse.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Un primo passo comune per implementare i controlli dei costi consiste nell'impostare delle notifiche quando si verificano eventi di costi o utilizzo al di fuori delle policy. In questo modo è possibile agire rapidamente e verificare se è necessaria un'azione correttiva, senza limitare o influire negativamente sui carichi di lavoro o sulle nuove attività. Una volta rilevati i limiti di carico di lavoro e ambiente, puoi applicare la governance. Budget AWS consente di impostare notifiche e definire budget mensili per i costi, l'utilizzo e gli sconti a fronte di impegni AWS (Savings Plans e istanze riservate). È possibile creare budget a livello di costo aggregato (ad esempio, tutti i costi) o a un livello più granulare, includendo solo dimensioni specifiche come account membri, servizi, tag o zone di disponibilità.

Una volta impostati i limiti di budget con Budget AWS, usa AWS Cost Anomaly Detection per ridurre i costi imprevisti. AWS Cost Anomaly Detection è un servizio di gestione dei costi che sfrutta il machine learning per il monitoraggio costante di costi e utilizzo al fine di rilevare spese anomale. Aiuta a individuare le spese anomale e le cause principali in modo da garantire un intervento tempestivo. Per prima cosa, crea un monitor dei costi in AWS Cost Anomaly Detection, quindi scegli le tue preferenze relativamente agli avvisi impostando una soglia in dollari (ad esempio, un avviso sulle spese anomale con impatto superiore a 1.000 dollari). Una volta ricevuti gli avvisi, puoi

analizzare la causa alla base dell'anomalia e l'impatto sui costi. Puoi inoltre monitorare ed eseguire la tua analisi delle anomalie in AWS Cost Explorer.

Applica le policy di governance in AWS tramite AWS Identity and Access Management e policy di controllo dei servizi (SCP) AWS Organizations. IAM consente di gestire in modo sicuro l'accesso ai servizi e alle risorse AWS. Utilizzando IAM, puoi controllare chi può creare e gestire le risorse di AWS, il tipo di risorse che possono essere create e dove possono essere create. In questo modo riduci al minimo la possibilità che vengano create risorse al di fuori della policy definita. Utilizza i ruoli e i gruppi creati in precedenza e assegna policy IAM per garantire l'utilizzo corretto. Le SCP offrono il controllo centralizzato sul numero massimo di autorizzazioni disponibili per tutti gli account nella tua organizzazione, assicurando che i tuoi account rimangano entro le linee guida di controllo degli accessi. Le SCP sono disponibili soltanto in un'organizzazione con tutte le funzionalità attivate e possono essere configurate in modo da rifiutare o consentire operazioni agli account membri per impostazione predefinita. Per ulteriori dettagli sull'implementazione della gestione degli accessi, consulta il whitepaper sul pilastro della sicurezza Well-Architected

La governance può essere implementata anche tramite la gestione delle quote di servizio di AWS. Assicurandoti che le quote di servizio siano impostate con spese minime e siano gestite in modo accurato, puoi ridurre al minimo la creazione di risorse che non rientrano nei requisiti della tua organizzazione. Per ottenere questo risultato, devi comprendere la velocità con cui i tuoi requisiti possono cambiare, valutare i progetti in corso (sia la creazione sia la disattivazione di risorse) e considerare la velocità con cui è possibile implementare le modifiche alle quote. Le quote di servizio possono essere utilizzate per aumentare le quote all'occorrenza.

Passaggi dell'implementazione

- Implementa notifiche sulle spese: tramite le policy aziendali definite, crea Budget AWS per ricevere notifiche quando le spese ricadono al di fuori delle tue policy. Configura più budget dei costi, uno per ogni account, in modo da ricevere informazioni sulla spesa complessiva del conto. Quindi configura budget di costo aggiuntivi all'interno di ciascun account per unità più piccole al suo interno. Queste unità variano a seconda della struttura dell'account. Alcuni esempi comuni sono Regioni AWS, carichi di lavoro (tramite i tag) o servizi AWS. Configura un elenco di distribuzione e-mail come destinatario per le notifiche e non un account e-mail di una singola persona. È possibile configurare un budget effettivo per quando un importo viene superato oppure utilizzare un budget previsto per la notifica dell'utilizzo previsto. Si possono anche preconfigurare operazioni di budget AWS, che possono applicare specifiche policy IAM o SCP o arrestare delle istanze Amazon EC2 o Amazon RDS definite. Le operazioni di budget possono essere avviate automaticamente o richiedere l'approvazione del flusso di lavoro.

- Implementa notifiche sulle spese anomale: usa AWS Cost Anomaly Detection per ridurre i costi imprevisti dell'organizzazione e analizzare la causa principale delle potenziali spese anomale. Una volta creato il sistema di monitoraggio dei costi per identificare le spese insolite alla granularità specificata e aver configurato le notifiche in AWS Cost Anomaly Detection, viene inviato un avviso quando sono rilevate spese insolite. Questo ti permetterà di analizzare le cause alla base dell'anomalia e di valutarne l'impatto sui costi. Utilizza le categorie di costo AWS durante la configurazione di AWS Cost Anomaly Detection per identificare il team di progetto o il team della business unit che può analizzare la causa principale del costo imprevisto e intraprendere tempestivamente le azioni necessarie.
- Implementa il controllo dell'utilizzo: utilizzando le policy dell'organizzazione definite, implementa ruoli e policy IAM per specificare quali operazioni possono o non possono eseguire gli utenti. In una policy AWS possono essere incluse più policy organizzative. Nello stesso modo in cui hai definito le policy, inizia in modo generale e quindi applica controlli più dettagliati a ogni fase. Anche le restrizioni dei servizi sono un controllo efficace sull'utilizzo. Implementa le restrizioni dei servizi corrette su tutti gli account.

Risorse

Documenti correlati:

- AWS managed policies for job functions
- AWS Strategia di fatturazione con account multipli di
- Controllo dell'accesso a Regioni AWS utilizzando le policy IAM
- Budget AWS
- AWS Cost Anomaly Detection
- Controllo dei costi AWS

Video correlati:

- How can I use Budget AWS to track my spending and usage

Esempi correlati:

- Example IAM access management policies
- Example service control policies

- Operazioni di AWS Budgets
- Creazione di una policy IAM per controllare l'accesso alle risorse Amazon EC2 utilizzando i tag
- Limita l'accesso di IAM Identity a risorse Amazon EC2 specifiche
- Integrazioni Slack per Cost Anomaly Detection mediante Amazon Q Developer nelle applicazioni di chat

COST02-BP06 Tieni traccia del ciclo di vita del progetto

Rileva, misura e controlla il ciclo di vita di progetti, team e ambienti per evitare di usare risorse non necessarie e pagare per esse.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Monitorando efficacemente il ciclo di vita del progetto, le organizzazioni possono ottimizzare il controllo dei costi attraverso una migliore pianificazione, gestione e ottimizzazione delle risorse. Gli approfondimenti acquisiti attraverso il monitoraggio sono preziosi per la formulazione di decisioni informate che contribuiscono alla competitività a livello di costi e al successo complessivo del progetto.

Il monitoraggio dell'intero ciclo di vita del carico di lavoro consente di capire quando i carichi di lavoro o i suoi componenti non sono più necessari. I carichi di lavoro e i componenti esistenti possono sembrare in uso, ma quando vengono AWS rilasciati nuovi servizi o funzionalità, possono essere smantellati o adottati. Controlla le fasi precedenti dei carichi di lavoro. Quando un carico di lavoro arriva in produzione, gli ambienti precedenti possono essere disattivati o notevolmente ridotti in termini di capacità fino a quando non sono nuovamente necessari.

Puoi applicare i tag alle risorse con un intervallo di tempo o un promemoria per aggiungere l'ora in cui il carico di lavoro è stato esaminato. Ad esempio, se sono trascorsi mesi dall'ultima volta che l'ambiente di sviluppo è stato esaminato, potrebbe essere il momento giusto per esaminarlo nuovamente per verificare se è possibile adottare nuovi servizi o se l'ambiente è in uso. È possibile raggruppare e contrassegnare le applicazioni con myApplicationson AWS per gestire e tenere traccia di metadati quali criticità, ambiente, ultima revisione e centro di costo. Puoi tenere traccia del ciclo di vita del carico di lavoro e monitorare e gestire i costi, lo stato di integrità, il livello di sicurezza e le prestazioni delle applicazioni.

AWS fornisce vari servizi di gestione e governance che è possibile utilizzare per il monitoraggio del ciclo di vita delle entità. È possibile utilizzare il AWS Confignostro AWS Systems Manager per fornire

un inventario dettagliato delle AWS risorse e della configurazione. Ti consigliamo di integrare questi servizi con i sistemi di gestione di progetti o asset esistenti per tenere traccia dei progetti attivi e dei prodotti all'interno della tua organizzazione. La combinazione del sistema attuale con il ricco set di eventi e metriche fornito da AWS consente di creare una visione degli eventi significativi del ciclo di vita e di gestire in modo proattivo le risorse per ridurre i costi non necessari.

Analogamente alla gestione del ciclo di vita delle applicazioni (ALM), il monitoraggio del ciclo di vita del progetto dovrebbe comportare la collaborazione di più processi, strumenti e team, ad esempio progettazione e sviluppo, test, produzione, supporto e ridondanza dei carichi di lavoro.

Monitorando attentamente ogni fase del ciclo di vita di un progetto, le organizzazioni ottengono informazioni cruciali e un maggiore controllo, semplificando la pianificazione, l'implementazione e la riuscita del progetto. Questa attenta supervisione verifica che i progetti non solo soddisfino gli standard di qualità, ma vengano consegnati in tempo e nel rispetto del budget, promuovendo l'efficienza complessiva dei costi.

Per ulteriori dettagli sull'implementazione del monitoraggio del ciclo di vita delle entità, consulta il whitepaper sul pilastro dell'eccellenza operativa di AWS Well-Architected.

Passaggi dell'implementazione

- Stabilisci il processo di monitoraggio del ciclo di vita del progetto: il team Centro di eccellenza del cloud deve predisporre un processo di monitoraggio del ciclo di vita del progetto. Stabilisci un approccio strutturato e sistematico per il monitoraggio dei carichi di lavoro al fine di migliorare il controllo, la visibilità e le prestazioni dei progetti. Rendi il processo di monitoraggio trasparente, collaborativo e incentrato sul miglioramento continuo per massimizzarne l'efficacia e il valore.
- Esegui le revisioni del carico di lavoro: in base a quanto definito dalle policy organizzative, stabilisci una cadenza regolare per l'audit dei progetti esistenti e le revisioni del carico di lavoro. L'impegno speso per il controllo deve essere proporzionale al rischio, al valore o al costo approssimativo per l'organizzazione. Le aree chiave da includere nell'audit sono il rischio di incidente o interruzione per l'organizzazione, il valore o contributo all'organizzazione (misurato in fatturato o reputazione del marchio), il costo del carico di lavoro (misurato come costo totale delle risorse e costi operativi) e l'utilizzo del carico di lavoro (misurato in numero di risultati dell'organizzazione per unità di tempo). Se queste aree cambiano durante il ciclo di vita, sono necessarie modifiche al carico di lavoro, ad esempio la disattivazione completa o parziale.

Risorse

Documenti correlati:

- Linee guida per l'etichettatura AWS
- Che cos'è ALM (Application Lifecycle Management)?
- AWS managed policies for job functions

Esempi correlati:

- Controlla l'accesso all'utilizzo delle politiche Regioni AWS IAM

Strumenti correlati

- AWS Config
- AWS Systems Manager
- Budget AWS
- AWS Organizations
- AWS CloudFormation

COST 3. In che modo monitori i costi e l'utilizzo?

Stabilisci policy e procedure per monitorare e allocare i costi in modo appropriato. Ciò ti permette di misurare e migliorare l'efficienza in termini di costi del carico di lavoro.

Best practice

- COST03-BP01 Configurazione di fonti di informazione dettagliate
- COST03-BP02 Aggiunta di informazioni sull'organizzazione a costi e utilizzo
- COST03-BP03 Identificazione delle categorie di attribuzione dei costi
- COST03-BP04 Stabilire metriche organizzative
- COST03-BP05 Configurazione degli strumenti di fatturazione e di gestione dei costi
- COST03-BP06 Allocazione dei costi in base alle metriche del carico di lavoro

COST03-BP01 Configurazione di fonti di informazione dettagliate

Imposta gli strumenti di gestione e report dei costi per ottenere una migliore analisi e trasparenza dei dati sui costi e sull’utilizzo. Configura il carico di lavoro per creare voci di log che facilitino il monitoraggio e la segregazione dei costi e dell’utilizzo.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Informazioni dettagliate sulla fatturazione, come la granularità oraria negli strumenti di gestione dei costi, consentono alle organizzazioni di tenere traccia dei propri consumi con ulteriori dettagli e di aiutarle a identificare alcuni dei motivi di aumento dei costi. Queste origini dati forniscono la visualizzazione più accurata dei costi e dell'utilizzo dell'intera organizzazione.

Puoi usare Esportazioni di dati AWS per creare esportazioni di AWS Cost and Usage Report (CUR) 2.0. Si tratta del nuovo modo consigliato per ricevere dati dettagliati su costi e utilizzo da AWS. Fornisce una granularità di utilizzo giornaliera o oraria, tariffe, costi e attributi di utilizzo per tutti i servizi AWS a pagamento (le stesse informazioni del CUR), oltre ad alcuni miglioramenti. Nel CUR sono inclusi tutti gli aspetti possibili, compresi tag, posizione, attributi delle risorse e ID account.

Esistono tre tipi di esportazione in base a ciò che desideri creare: l'esportazione di dati standard, l'esportazione in una dashboard costi e utilizzo con l'integrazione di QuickSight oppure l'esportazione di dati legacy.

- Esportazione dati standard: esportazione personalizzata di una tabella distribuita su Amazon S3 su base ricorrente.
- Dashboard costi e utilizzo: esportazione e integrazione in QuickSight per implementare una dashboard costi e utilizzo precostituita.
- Esportazione di dati legacy: esportazione dell'AWS Cost and Usage Report (CUR) legacy.

È possibile creare esportazioni di dati con le seguenti personalizzazioni:

- Inclusione degli ID risorsa
- Dati di allocazione dei costi suddivisi
- Granularità oraria
- Controllo delle versioni
- Tipo di compressione e formato del file

Per i carichi di lavoro che eseguono container su Amazon ECS o Amazon EKS, abilita i dati di allocazione dei costi suddivisi in modo da poter allocare i costi dei container a singole business unit e team, in base al modo in cui i carichi di lavoro dei container consumano le risorse di calcolo

e memoria condivise. I dati di allocazione dei costi suddivisi introducono in AWS Cost and Usage Report i dati sui costi e sull'utilizzo per le nuove risorse a livello di container. I dati di allocazione dei costi suddivisi vengono calcolati determinando il costo di singoli servizi e attività ECS in esecuzione sul cluster.

La dashboard costi e utilizzo esporta la tabella dei costi e dell'utilizzo in un bucket S3 su base ricorrente e implementa una dashboard costi e utilizzo predefinita in QuickSight. Utilizza questa opzione se desideri implementare rapidamente una dashboard dei dati su costi e utilizzo senza funzionalità di personalizzazione.

Se lo desideri, puoi comunque esportare CUR in modalità legacy per integrare altri servizi di elaborazione, come AWS Glue, al fine di preparare i dati per l'analisi ed eseguirla con Amazon Athena mediante SQL per le query sui dati.

Passaggi dell'implementazione

- Crea esportazioni di dati: crea esportazioni personalizzate con i dati che desideri e controlla lo schema delle tue esportazioni. Crea le esportazioni dei dati di fatturazione e gestione dei costi utilizzando SQL di base e visualizza i dati di fatturazione e gestione dei costi integrandoli con QuickSight. Puoi anche esportare i dati in modalità standard per analizzarli con altri strumenti di elaborazione, come Amazon Athena.
- Configura il report su costi e utilizzo: utilizzando la console di fatturazione, configura almeno un report costi e utilizzo. Configura un report con granularità oraria che include tutti gli identificatori e gli ID risorsa. Puoi anche creare altri report con granularità diverse per fornire informazioni di riepilogo di livello superiore.
- Configura la granularità oraria in Cost Explorer: per accedere ai dati su costi e utilizzo con granularità oraria negli ultimi 14 giorni, prendi in considerazione l'abilitazione di dati a livello di ora e risorsa nella console di fatturazione.
- Configura la registrazione dei log da parte delle applicazioni: verifica che l'applicazione registri ogni risultato aziendale che distribuisce in modo che possa essere monitorato e misurato. Assicurati che la granularità di questi dati sia almeno oraria, affinché possa essere abbinata ai dati relativi a costi e utilizzo. Per maggiori dettagli su creazione di log e monitoraggio, consulta il pilastro dell'eccellenza operativa Well-Architected.

Risorse

Documenti correlati:

- Esportazioni di dati AWS
- AWS Glue
- Rapidità
- AWS Cost Management Pricing
- Applicazione di tag alle risorse AWS
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

Esempi correlati:

- AWS Configurazione dell’account
- Data Exports for AWS Billing and Cost Management
- AWS Cost Explorer Casi di utilizzo comune

COST03-BP02 Aggiunta di informazioni sull'organizzazione a costi e utilizzo

Definisci uno schema per l'applicazione di tag in base alla tua organizzazione, agli attributi del carico di lavoro e alle categorie di allocazione dei costi, in modo da poter filtrare e cercare le risorse o monitorare costi e utilizzo negli strumenti di gestione dei costi. Implementa un'applicazione di tag consistente in tutte le risorse possibili per scopo, team, ambiente o altri criteri pertinenti alla tua azienda.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Implementa l'applicazione di tag AWS in per aggiungere informazioni sull'organizzazione alle risorse, che verranno, quindi, integrate alle informazioni su costi e utilizzo. Un tag è una coppia chiave-valore: la chiave è definita e deve essere univoca all'interno dell'organizzazione, mentre il valore è univoco per un gruppo di risorse. Ad esempio, una coppia chiave-valore può essere costituita da `Environment`, con un valore di `Production`. Tutte le risorse nell'ambiente di produzione avranno questa coppia chiave-valore. L'applicazione di tag consente di categorizzare e monitorare i costi con informazioni significative e rilevanti sull'organizzazione. Puoi applicare tag che rappresentano categorie dell'organizzazione (ad esempio, centri di costo, nomi di applicazioni, progetti o proprietari) e identificano carichi di lavoro e rispettive funzionalità (come test o produzione) per attribuire i costi e l'utilizzo all'interno dell'organizzazione.

Quando applichi i tag alle tue risorse AWS (come le istanze Amazon Elastic Compute Cloud o i bucket Amazon Simple Storage Service) e li attivi, AWS aggiunge queste informazioni ai report su costi e utilizzo. Puoi creare report e condurre analisi su risorse con tag e senza tag per incrementare la conformità con le policy di gestione dei costi interne e garantire un'attribuzione accurata.

La creazione e l'implementazione di uno standard per l'applicazione di tag AWS tra gli account dell'organizzazione agevola la gestione e amministrazione degli ambienti AWS in modo coerente e uniforme. Usa le policy di tag in AWS Organizations per definire regole su come i tag possono essere applicati alle risorse AWS nei tuoi account in AWS Organizations. Le policy di tag consentono di adottare con facilità un approccio standardizzato per l'applicazione di tag alle risorse AWS.

AWS Tag Editor consente di aggiungere, eliminare e gestire tag di più risorse. Con questa funzionalità, è possibile cercare le risorse a cui applicare tag e quindi gestirli per quelle risorse dei tuoi risultati di ricerca.

Le categorie di costo AWS consentono di assegnare ai tuoi costi significati per l'organizzazione, senza necessità di applicare tag alle risorse. Puoi mappare le informazioni su costi e utilizzo attribuendole a strutture organizzative interne univoche. Puoi definire regole di categoria per mappare e categorizzare i costi utilizzando le dimensioni di fatturazione, ad esempio account e tag. Questo offre un altro livello di funzionalità di gestione oltre all'applicazione di tag. Puoi anche mappare account e tag specifici attribuendoli a più progetti.

Passaggi dell'implementazione

- Definisci uno schema per l'applicazione di tag: riunisci tutte le parti interessate dell'azienda per definire uno schema. Questo generalmente include i ruoli tecnici, finanziari e di gestione. Definisci un elenco di tag che tutte le risorse devono avere, nonché un elenco di tag che le risorse dovrebbero avere. Verifica che i nomi e i valori dei tag siano coerenti all'interno dell'organizzazione.
- Risorse di tag: utilizzando le categorie di attribuzione dei costi definite, posiziona i tag in tutte le risorse dei carichi di lavoro in base alle categorie. Utilizza strumenti come l'interfaccia a riga di comando (CLI), Tag Editor o AWS Systems Manager per aumentare l'efficienza.
- Implementa le categorie di costo AWS: puoi creare categorie di costo senza implementare l'applicazione di tag. Le categorie di costo utilizzano le dimensioni di costo e utilizzo esistenti. Crea regole di categoria dallo schema e implementale nelle categorie di costo.
- Automatizza l'applicazione dei tag: per verificare di mantenere elevati livelli di applicazione di tag tra tutte le risorse, automatizza l'applicazione di tag in modo che le risorse siano contrassegnate automaticamente al momento della creazione. Utilizza servizi come AWS CloudFormation per verificare l'avvenuta applicazione di tag alle risorse al momento della creazione. Puoi anche creare

una soluzione personalizzata per l'applicazione in automatico di tag mediante le funzioni Lambda o usare un microservizio che scansioni periodicamente il carico di lavoro e rimuova le risorse prive di tag, l'ideale per ambienti di test e sviluppo.

- Monitora ed elabora report sull'applicazione di tag: per verificare di mantenere elevati livelli di applicazione di tag nella tua organizzazione, elabora report e monitora i tag tra i tuoi carichi di lavoro. Puoi utilizzare AWS Cost Explorer per visualizzare il costo delle risorse con tag e senza tag oppure utilizzare servizi come Tag Editor. Verifica regolarmente il numero di risorse senza tag e aggiungi i tag fino a raggiungere il livello desiderato di applicazione di tag.

Risorse

Documenti correlati:

- Tagging Best Practices
- AWS CloudFormation Resource Tag
- AWS Cost Categories
- Applicazione di tag alle risorse AWS
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Gestione del report su costi e utilizzo di AWS

Video correlati:

- How can I tag my AWS resources to divide up my bill by cost center or project
- Tagging AWS Resources

COST03-BP03 Identificazione delle categorie di attribuzione dei costi

Identifica le categorie dell'organizzazione, come business unit, reparti o progetti, che potrebbero essere utilizzate per allocare i costi alle entità responsabili dei consumi interni. Utilizza queste categorie per rafforzare la responsabilità della spesa, creare consapevolezza dei costi e promuovere comportamenti di consumo efficaci.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Il processo di categorizzazione dei costi è fondamentale nella definizione del budget, nella contabilità, nella rendicontazione finanziaria, nel processo decisionale, nell'analisi comparativa e nella gestione dei progetti. La categorizzazione e la classificazione delle spese consentono ai team di comprendere meglio i tipi di costi che dovranno sostenere durante il loro percorso verso il cloud, aiutandoli a prendere decisioni informate e a gestire i budget in modo efficace.

La responsabilità della spesa cloud costituisce un forte incentivo per una gestione disciplinata della domanda e dei costi. Il risultato è un notevole risparmio sui costi del cloud per le organizzazioni che destinano la maggior parte della loro spesa per il cloud a business unit o team che utilizzano il cloud. Inoltre, l'allocazione della spesa per il cloud aiuta le organizzazioni ad adottare un numero maggiore di best practice di governance del cloud centralizzate.

Collabora con il tuo team finanziario e altre parti interessate per comprendere i requisiti di allocazione dei costi all'interno della tua organizzazione durante le chiamate organizzate con periodicità regolare. I costi del carico di lavoro devono essere allocati per tutto il ciclo di vita, inclusi sviluppo, test, produzione e disattivazione. Comprendi in che modo i costi sostenuti per formazione, sviluppo del personale e creazione di idee sono attribuiti all'interno dell'organizzazione. Questo può essere utile per assegnare correttamente gli account utilizzati a questo scopo ai budget di formazione e sviluppo anziché ai budget di costi IT generici.

Una volta definite le categorie di attribuzione dei costi con le parti interessate dell'organizzazione, utilizza le categorie di costo AWS per raggruppare le informazioni su costi e utilizzo in categorie significative nell'Cloud AWS, ad esempio il costo di un progetto specifico o gli Account AWS per reparti o business unit. Puoi creare categorie personalizzate e mappare le informazioni su costi e utilizzo in queste categorie in base a regole che definisci usando componenti diversi come account, tag, servizio o tipo di addebito. Una volta impostate le categorie di costi, è possibile visualizzare le informazioni su costi e utilizzo consentendo così alla tua organizzazione di prendere decisioni di acquisto e strategiche migliori. Tali categorie sono visibili in AWS Cost Explorer, Budget AWS e anche in AWS Cost and Usage Report.

Ad esempio, crea categorie di costo per le tue business unit (team DevOps) e in ogni categoria crea più regole (regole per ogni sottocategoria) con più componenti (Account AWS, tag di allocazione dei costi, servizi o tipo di addebito) in base ai raggruppamenti da te definiti. Con le categorie di costo puoi organizzare i costi utilizzando un motore basato su regole. Le regole configurate organizzeranno i costi in categorie. All'interno di queste regole, puoi filtrare utilizzando più aspetti o componenti per ciascuna categoria, come Account AWS, servizi AWS o tipi di addebito specifici. È possibile

utilizzare queste categorie tra più prodotti nelle console di Gestione dei costi e fatturazione AWS e Cost Management. Sono inclusi AWS Cost Explorer, Budget AWS, AWS Cost and Usage Report e AWS Cost Anomaly Detection.

Come esempio, il diagramma seguente mostra in che modo raggruppare i costi e le informazioni sull'utilizzo nella tua organizzazione definendo più team (categoria di costo), molteplici ambienti (regole) e assegnando a ogni ambiente molteplici risorse o asset (dimensioni).

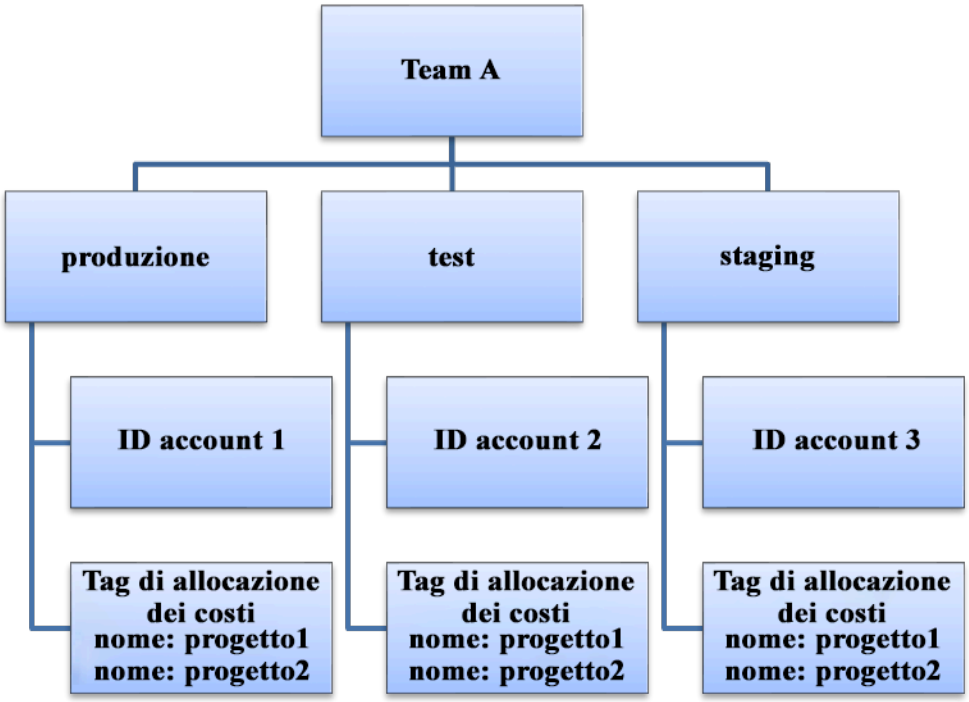


Diagramma relativo a costi e utilizzo

Puoi anche creare gruppi di costi con le categorie di costo. Dopo aver creato le categorie di costo (attendi fino a 24 ore dopo la creazione di una categoria di costo affinché i record di utilizzo siano aggiornati con i valori), verranno visualizzati in AWS Cost Explorer, Budget AWS, AWS Cost and Usage Report e AWS Cost Anomaly Detection. In AWS Cost Explorer e Budget AWS, una categoria di costo appare come una componente di fatturazione aggiuntiva. Puoi utilizzare questa opzione per filtrare il valore specifico della categoria di costo o raggruppare in base alla categoria di costo.

Passaggi dell'implementazione

- Definisci le categorie dell'organizzazione: organizza riunioni con le parti interessate interne e le business unit per definire categorie che riflettano la struttura e i requisiti della tua organizzazione. Queste categorie dovrebbero corrispondere direttamente alla struttura delle categorie finanziarie

esistenti, ad esempio business unit, budget, centro di costi o reparto. Osserva i risultati che il cloud offre per la tua azienda, ad esempio la formazione o l'istruzione, poiché anche queste sono categorie organizzative.

- Definisci le categorie funzionali: organizza riunioni con le parti interessate interne e le unità di business per definire categorie che riflettano le funzioni presenti all'interno della tua azienda. Potrebbe trattarsi del carico di lavoro o dei nomi delle applicazioni e del tipo di ambiente, ad esempio produzione, test o sviluppo.
- Definisci le categorie di costo AWS: crea categorie di costo in modo da organizzare le informazioni su costo e utilizzo mediante le categorie di costo AWS e mappare costo e uso di AWS in categorie significative. A una risorsa possono essere assegnate più categorie e una risorsa può essere in più categorie diverse, quindi definisci tutte le categorie necessarie in modo da essere in grado di gestire i tuoi costi all'interno delle strutture categorizzate mediante le categorie di costo AWS.

Risorse

Documenti correlati:

- Applicazione di tag alle risorse AWS
- Utilizzo dei tag per l'allocazione dei costi
- Analisi dei costi con Budget AWS
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports
- AWS Cost Categories
- Managing your costs with AWS Cost Categories
- Creating cost categories
- Tagging cost categories
- Splitting charges within cost categories
- Funzionalità delle categorie di costo AWS

Esempi correlati:

- Organize your cost and usage data with AWS Cost Categories
- Managing your costs with AWS Cost Categories

COST03-BP04 Stabilire metriche organizzative

Definisci i parametri dell'organizzazione necessari per il carico di lavoro. I parametri esemplificativi di un carico di lavoro sono i report dei clienti prodotti o le pagine Web scaricate dai clienti.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Comprendi in che modo viene misurato l'output del carico di lavoro rispetto al successo aziendale. Ogni carico di lavoro ha in genere un piccolo set di output principali che indicano le prestazioni. Se disponi di un carico di lavoro complesso con molti componenti, puoi dare priorità alle voci dell'elenco o definire e monitorare i parametri per ogni componente. Collabora con i tuoi team per capire quali parametri utilizzare. Questa unità verrà utilizzata per comprendere l'efficienza del carico di lavoro o il costo per ciascun output aziendale.

Passaggi dell'implementazione

- Definisci i risultati del carico di lavoro: organizza riunioni con le parti interessate dell'azienda e definisci i risultati del carico di lavoro. Si tratta di una misura principale dell'utilizzo da parte dei clienti e devono essere parametri aziendali e non parametri tecnici. Deve esserci un piccolo numero di parametri di alto livello (meno di cinque) per carico di lavoro. Se il carico di lavoro produce più risultati per diversi casi d'uso, raggruppali in un singolo parametro.
- Definisci i risultati dei componenti del carico di lavoro: facoltativamente, se disponi di un carico di lavoro grande e complesso oppure puoi suddividere facilmente il carico di lavoro in componenti (ad esempio microservizi) con input e output ben definiti, definisci i parametri per ogni componente. Lo sforzo deve riflettere il valore e il costo del componente. Inizia con i componenti più grandi e punta ai componenti più piccoli.

Risorse

Documenti correlati:

- Etichettare AWS le risorse
- Analisi dei costi con Budgets AWS
- Analyzing your costs with Cost Explorer
- Gestione del report su costi e utilizzo di AWS

COST03-BP05 Configurazione degli strumenti di fatturazione e di gestione dei costi

Configura gli strumenti di gestione dei costi in conformità alle policy della tua organizzazione per gestire e ottimizzare gli investimenti nel cloud. Sono inclusi servizi, strumenti e risorse per organizzare e monitorare i dati su costi e utilizzo, migliorare il controllo tramite la fatturazione consolidata e le autorizzazioni di accesso, perfezionare la pianificazione tramite budget e previsioni, ricevere notifiche o avvisi e ridurre i costi tramite l'ottimizzazione di prezzi e risorse.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per definire consapevolezza e responsabilità forti, la strategia che interessa l'account deve essere considerata come parte integrante della strategia di allocazione dei costi. Definisci questo concetto ora per non doverlo affrontare in futuro. In caso contrario, il livello di consapevolezza potrebbe essere insufficiente e potrebbero verificarsi problemi in seguito.

Per incoraggiare la responsabilità degli investimenti nel cloud, fornisci agli utenti l'accesso a strumenti che offrono visibilità su costi e utilizzo. AWS consiglia di configurare tutti i carichi di lavoro e definire i team per i seguenti scopi:

- Organizzazione: definisci l'allocazione dei costi e i riferimenti della governance con la tua strategia di applicazione dei tag e la tassonomia. Crea più account AWS con strumenti come AWS Control Tower o AWS Organization. Applica i tag alle risorse AWS supportate e classificale in modo significativo in base alla struttura organizzativa (business unit, reparti o progetti). Applica i tag ai nomi degli account per centri di costo specifici e mappali con AWS Cost Categories per raggruppare gli account delle business unit nei relativi centri di costo in modo che il responsabile della business unit possa visualizzare il consumo di più account in un'unica posizione.
- Accesso: tieni traccia delle informazioni di fatturazione a livello di organizzazione nella fatturazione consolidata e verifica che le parti interessate e i responsabili idonei abbiano accesso.
- Controllo: crea meccanismi di governance efficaci con i giusti guardrail per prevenire scenari imprevisti quando utilizzi policy di controllo dei servizi, policy di tag, policy IAM e avvisi sul budget. Ad esempio, puoi consentire ai team di creare risorse specifiche nelle regioni preferite solo utilizzando meccanismi di controllo efficaci e impedire la creazione di risorse prive di tag specifici, come il centro di costo.
- Stato attuale: configura un pannello di controllo che mostra i livelli correnti di costi e utilizzo. Il pannello di controllo deve essere disponibile in un luogo altamente visibile all'interno dell'ambiente di lavoro, in modo simile al pannello di controllo delle operazioni. Puoi esportare i dati e utilizzare la

Dashboard costi e utilizzo dalla Centrale ottimizzazione costi AWS o qualsiasi prodotto supportato per creare questa visibilità. Potresti dover creare pannelli di controllo diversi per tipi di utenti diversi, ad esempio il pannello di controllo per i manager sarà diverso da quello di progettazione.

- Notifiche: invia notifiche in caso di superamento dei limiti definiti in termini di costo o utilizzo e anomalie con AWS Budgets o AWS Cost Anomaly Detection.
- Report: riepiloga tutte le informazioni su costi e utilizzo. Aumenta la consapevolezza e la responsabilità dei tuoi investimenti nel cloud con dati sui costi dettagliati e attribuibili. Crea i report con i suggerimenti pertinenti per il team che li utilizza.
- Monitoraggio: mostra i costi e l'utilizzo attuali rispetto a obiettivi o target stabiliti.
- Analisi: offri ai membri del team la possibilità di eseguire analisi personalizzate e approfondite fino alla granularità oraria, giornaliera o mensile con diversi filtri (risorse, account, tag, ecc.).
- Esame: non perdere gli aggiornamenti sulle opportunità di implementazione delle risorse e di ottimizzazione dei costi. Ricevi le notifiche utilizzando Amazon CloudWatch, Amazon SNS o Amazon SES per le implementazioni delle risorse a livello di organizzazione. Esamina i suggerimenti per l'ottimizzazione dei costi con AWS Trusted Advisor o AWS Compute Optimizer.
- Report delle tendenze: mostra la variabilità dei costi e dell'utilizzo nel periodo richiesto e con la granularità richiesta.
- Previsioni: mostra i costi futuri stimati, prevedi l'utilizzo delle risorse e investi con pannelli di controllo di previsioni che tu stesso crei.

Sfrutta la Centrale ottimizzazione costi AWS per esaminare da una posizione centralizzata le potenziali opportunità di risparmio sui costi consolidati, nonché per creare esportazioni di dati per l'integrazione con Amazon Athena. Puoi utilizzare la Centrale ottimizzazione costi AWS anche per implementare la Dashboard costi e utilizzo, che usa QuickSight per l'analisi interattiva dei costi e la condivisione sicura degli approfondimenti sui costi.

Se non disponi delle competenze o della larghezza di banda essenziali nella tua organizzazione, puoi usare AWS ProServ, AWS Managed Services (AMS) o rivolgerti ai partner AWS. Puoi anche utilizzare strumenti di terze parti, ma assicurati di convalidare la proposta di valore.

Passaggi dell'implementazione

- Consenti l'accesso agli strumenti in base ai team: configura i tuoi account e crea gruppi con accesso ai report su costi e utilizzo necessari per i loro consumi e usa AWS Identity and Access Management per controllare l'accesso a strumenti come AWS Cost Explorer. Questi gruppi devono includere i rappresentanti di tutti i team che possiedono o gestiscono un'applicazione. In questo

modo si certifica che ogni team ha accesso alle informazioni sui costi e sull'utilizzo per tenere traccia dei propri consumi.

- Organizza tag e categorie di costo: organizza i costi tra team, business unit, applicazioni, ambienti e progetti. Usa i tag delle risorse per organizzare i costi, in base ai tag di allocazione dei costi. Crea le categorie di costo in base alle dimensioni utilizzando tag, account, servizi e così via per mappare i costi.
- Configura AWS Budgets: configura AWS Budgets in tutti gli account del carico di lavoro. Imposta un budget per la spesa complessiva dell'account e un budget per il carico di lavoro utilizzando i tag e le categorie di costo. Configura le notifiche in AWS Budgets per ricevere allarmi quando superi gli importi previsti nel budget o quando i costi stimati sono superiori a quelli dei tuoi budget.
- Configura AWS Cost Anomaly Detection: usa AWS Cost Anomaly Detection per i tuoi account, i servizi di base o le categorie di costo che hai creato per monitorare costi e utilizzo e individuare investimenti insoliti. Puoi ricevere avvisi individualmente in report aggregati, oppure avvisi in un'email o in un argomento Amazon SNS per poter analizzare e stabilire il motivo principale di un'anomalia, nonché identificare il fattore che determina l'aumento dei costi.
- Usa gli strumenti di analisi dei costi: configura AWS Cost Explorer per il tuo carico di lavoro e gli account e visualizza i dati sui costi per ulteriori analisi. Crea un pannello di controllo per il carico di lavoro che tenga traccia della spesa generale e le metriche di utilizzo chiave per il carico di lavoro, nonché preveda i costi futuri sulla base dei tuoi dati storici.
- Utilizza strumenti di analisi per il risparmio sui costi: usa il Centrale ottimizzazione costi AWS per individuare i possibili risparmi con suggerimenti personalizzati, tra cui l'eliminazione delle risorse inutilizzate, il ridimensionamento corretto, i Savings Plans, le prenotazioni e i suggerimenti per l'ottimizzazione del calcolo.
- Configura strumenti avanzati: puoi creare in modo facoltativo immagini per agevolare l'analisi interattiva e la condivisione delle informazioni sui costi. Con le esportazioni dei dati della Centrale ottimizzazione costi AWS puoi creare per l'organizzazione una Dashboard costi e utilizzo basata su QuickSight con dettagli e granularità aggiuntivi. Puoi anche implementare funzionalità di analisi avanzate tramite le esportazioni di dati in Amazon Athena per query avanzate e creare pannelli di controllo su QuickSight. Collabora con i partner AWS per adottare soluzioni di gestione del cloud per il monitoraggio e l'ottimizzazione della fatturazione consolidata del cloud.

Risorse

Documenti correlati:

- What is Gestione dei costi e fatturazione AWS and Cost Management?

- Stabilire il tuo ambiente AWS per le best practice
- Best Practices for Tagging AWS Resources
- Tagging delle risorse AWS
- AWS Cost Categories
- Analyzing your costs with AWS Budgets
- Analisi dei costi con AWS Cost Explorer
- What is AWS Data Exports?

Video correlati:

- Deploying Cloud Intelligence Dashboards
- Get Alerts on any FinOps or Cost Optimization Metric or KPI

Esempi correlati:

- Dashboard costi e utilizzo con tecnologia Quick
- Workshop su AWS Cost and Usage Governance

COST03-BP06 Allocazione dei costi in base alle metriche del carico di lavoro

Alloca i costi del carico di lavoro in base alle metriche di utilizzo o ai risultati aziendali per misurare l'efficienza dei costi del carico di lavoro. Implementa un processo per analizzare i dati relativi a costi e utilizzo con i servizi di analisi, che possono fornire informazioni approfondite e funzionalità di chargeback.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Ottimizzare i costi significa conseguire i risultati aziendali al prezzo più basso eseguendo l'allocazione dei costi del carico di lavoro in base alle metriche di quest'ultimo, misurate in termini di efficienza. Monitora le metriche del carico di lavoro definite tramite file di log o altre funzionalità di monitoraggio dell'applicazione. Combina questi dati con i costi del carico di lavoro, che possono essere ottenuti osservando i costi con un determinato valore di tag o ID account. Esegui questa analisi a livello orario. L'efficienza cambia in genere se disponi di componenti di costo statico, come un database backend sempre in esecuzione, con un tasso di richiesta variabile, ad esempio picchi di utilizzo tra

le 9:00 e le 17:00 con poche richieste di notte. Comprendere la relazione tra i costi statici e i costi variabili ti aiuterà a rendere più mirate le tue attività di ottimizzazione.

La creazione di parametri del carico di lavoro per risorse condivise può essere difficile rispetto a risorse come applicazioni containerizzate su Amazon Elastic Container Service (Amazon ECS) e Gateway Amazon API. Tuttavia, esistono alcuni modi per classificare l'utilizzo e tenere traccia dei costi. Se devi monitorare le risorse condivise di AWS Batch e Amazon ECS, puoi abilitare i dati di allocazione dei costi suddivisi in AWS Cost Explorer. Con i dati di allocazione dei costi suddivisi, puoi analizzare e ottimizzare i costi e l'utilizzo delle tue applicazioni containerizzate e riallocare i costi delle applicazioni alle singole entità aziendali in base al modo in cui vengono consumate le risorse di calcolo e memoria condivise.

Passaggi dell'implementazione

- Alloca i costi alle metriche del carico di lavoro: utilizzando le metriche e l'applicazione di tag definiti e configurati, crea una metrica che combini l'output e il costo del carico di lavoro. Utilizza i servizi di analisi come Amazon Athena e Amazon Quick per creare un pannello di controllo di efficienza per il carico di lavoro complessivo e qualsiasi componente.

Risorse

Documenti correlati:

- Applicazione di tag alle risorse AWS
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Gestione del report su costi e utilizzo di AWS

Esempi correlati:

- Improve cost visibility of Amazon ECS and AWS Batch with AWS Split Cost Allocation Data

COST 4. In che modo disattivi le risorse?

Implementa il controllo del cambiamento e la gestione delle risorse dall'inizio del progetto alla fine del ciclo di vita. In questo modo, puoi disattivare o terminare le risorse non utilizzate per ridurre gli sprechi.

Best practice

- COST04-BP01 Tieni traccia delle risorse per tutto il loro ciclo di vita
- COST04-BP02 Implementazione di un processo di disattivazione
- COST04-BP03 Disattivazione delle risorse
- COST04-BP04 Disattivazione automatica delle risorse
- COST04-BP05 Applicare policy di conservazione dei dati

COST04-BP01 Tieni traccia delle risorse per tutto il loro ciclo di vita

Definisci e implementa un metodo per monitorare le risorse e le loro associazioni con i sistemi durante il loro ciclo di vita. Puoi usare l'applicazione di tag per identificare il carico di lavoro o la funzione della risorsa.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Disattiva le risorse dei carichi di lavoro che non sono più necessarie. Un esempio comune sono le risorse utilizzate per i test: dopo il completamento dei test, le risorse possono essere rimosse. La tracciabilità delle risorse con i tag (e la predisposizione di report su tali tag) può aiutare a identificare le risorse da disattivare, poiché non saranno più in uso o la loro licenza è in scadenza. L'utilizzo dei tag è un modo efficace per monitorare le risorse: puoi etichettare la risorsa con la relativa funzione o con una data nota in cui può essere disattivata. Puoi quindi eseguire i report su questi tag. Esempi di valori per l'applicazione di tag relativi alle funzionalità sono `feature-X testing` per identificare lo scopo della risorsa in termini di ciclo di vita del carico di lavoro. Un altro esempio è l'utilizzo di `LifeSpan` o `TTL` per le risorse, ad esempio il nome della chiave e il valore del to-be-deleted tag per definire il periodo di tempo o l'ora specifica per la disattivazione.

Passaggi dell'implementazione

- Implementa uno schema di applicazione di tag: implementa uno schema di applicazione di tag che identifichi il carico di lavoro a cui appartiene la risorsa, verificando che tutte le risorse all'interno del carico di lavoro siano contrassegnate di conseguenza. L'applicazione dei tag aiuta a classificare le risorse in base allo scopo, al team, all'ambiente o ad altri criteri rilevanti per l'azienda. Per ulteriori informazioni su casi d'uso, strategie e tecniche di applicazione dei tag, consulta AWS Tagging Best Practices.
- Implementa il monitoraggio di throughput del carico di lavoro o output: implementa il monitoraggio degli allarmi del throughput del carico di lavoro, avviandolo per le richieste di input o i

completamenti dell'output. Configuralo per fornire notifiche quando le richieste o gli output del carico di lavoro scendono a zero, indicando che le risorse del carico di lavoro non sono più utilizzate. Incorpora un fattore temporale se il carico di lavoro scende periodicamente a zero in condizioni normali. Per maggiori informazioni sulle risorse inutilizzate o sottoutilizzate, consulta Controlli dell'ottimizzazione dei costi AWS Trusted Advisor.

- AWS Risorse di gruppo: crea gruppi per le AWS risorse. È possibile utilizzare AWS Resource Groupsper organizzare e gestire le AWS risorse che si trovano all'interno delle stesse Regione AWS. Puoi aggiungere tag alla maggior parte delle risorse affinché sia possibile identificarle e ordinarle all'interno dell'organizzazione. Usa l'editor di tag per aggiungere tag alle risorse supportate in blocco. Prendi in considerazione l'utilizzo di AWS Service Catalog per creare, gestire e distribuire agli utenti finali portafogli di prodotti approvati e gestire il ciclo di vita del prodotto.

Risorse

Documenti correlati:

- AWS Auto Scaling
- AWS Trusted Advisor
- AWS Trusted Advisor Controlli di ottimizzazione dei costi
- Etichettatura delle risorse AWS
- Publishing Custom Metrics

Video correlati:

- Come ottimizzare i costi utilizzando AWS Trusted Advisor

Esempi correlati:

- Organizza AWS le risorse
- Ottimizza i costi utilizzando AWS Trusted Advisor

COST04-BP02 Implementazione di un processo di disattivazione

Implementa un processo per identificare e disattivare le risorse inutilizzate.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Implementa un processo standardizzato in tutta l’organizzazione per identificare e rimuovere le risorse inutilizzate. Il processo deve definire la frequenza di esecuzione della ricerca e i processi per rimuovere la risorsa al fine di verificare che tutti i requisiti dell’organizzazione siano soddisfatti.

Passaggi dell’implementazione

- Crea e implementa un processo di disattivazione: collaborando con sviluppatori e proprietari del carico di lavoro, crea un processo di disattivazione per il carico di lavoro e le relative risorse. Il processo deve includere il metodo per verificare se il carico di lavoro è in uso e quello per capire se ciascuna delle risorse del carico di lavoro è in uso. Specifica le fasi necessarie per disattivare la risorsa, rimuovendola dal servizio e garantendo allo stesso tempo la conformità a qualsiasi requisito normativo. Dovrebbero essere incluse tutte le risorse associate, come le licenze o lo spazio di archiviazione collegato. Invia una notifica ai proprietari del carico di lavoro indicando che il processo di disattivazione è stato avviato.

  Utilizza i seguenti passaggi di disattivazione per guidarti su quali dovrebbero essere le verifiche eseguite come parte del processo:

  - Identifica le risorse da disattivare: individua le risorse idonee alla disattivazione nel tuo ambiente Cloud AWS. Registra tutte le informazioni necessarie e pianifica la disattivazione. Nella sequenza temporale, assicurati di tenere conto di eventuali problemi imprevisti e di quando si verificano durante il processo.
  - Coordina e comunica: collabora con i proprietari dei carichi di lavoro per ricevere conferma circa le risorse da eliminare.
  - Registra metadati e crea backup: registra metadati (come IP pubblici, regioni, zone di disponibilità, VPC, sottoreti e gruppi di sicurezza) e crea backup (come snapshot di Amazon Elastic Block Store o AMI, esportazione di chiavi ed esportazione di certificati) se necessario per le risorse nell’ambiente di produzione o se si tratta di risorse critiche.
  - Convalida il modello Infrastructure as code: determina se le risorse sono state implementate con CloudFormation, Terraform, AWS Cloud Development Kit (AWS CDK) o qualsiasi altro strumento di implementazione Infrastructure as code in modo da implementale di nuovo, se necessario.
  - Impedisci l’accesso: applica controlli restrittivi per un periodo di tempo, in modo da impedire l’uso delle risorse mentre stabilisci se sono necessarie o meno. Verifica che l’ambiente delle risorse possa essere ripristinato allo stato originale, se necessario.
  - Segui il processo di disattivazione interno: segui le attività amministrative e il processo di disattivazione della tua organizzazione, come la rimozione della risorsa dal dominio

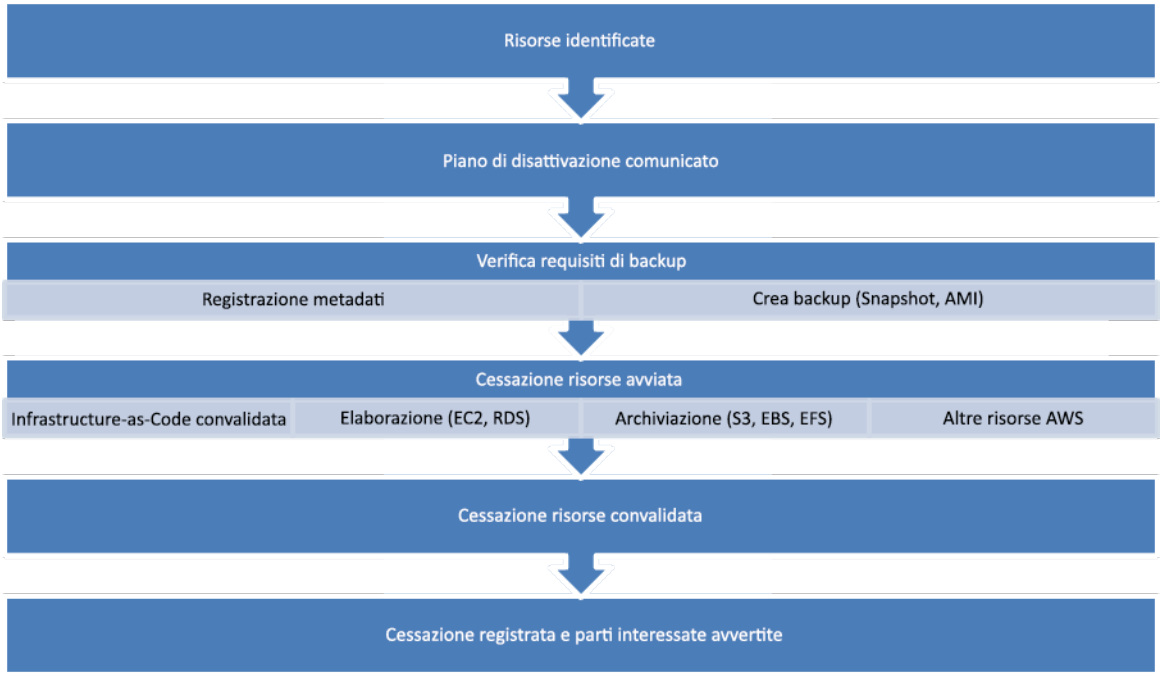
dell'organizzazione, la rimozione del record DNS e della risorsa dagli strumenti di gestione della configurazione, monitoraggio, di automazione e sicurezza.

Se la risorsa è un'istanza Amazon EC2, consulta l'elenco seguente. Per ulteriori dettagli, consulta Come posso eliminare o interrompere le mie risorse Amazon EC2?

- Arresta o interrompi tutti i bilanciatori del carico e le istanze Amazon EC2. Le istanze Amazon EC2 sono mostrate nella console per un breve periodo prima di essere terminate. Non verrà addebitato alcun costo per le istanze che non si trovano in stato di esecuzione
- Elimina la tua infrastruttura Auto Scaling.
- Rilascia tutti gli host dedicati.
- Elimina tutti i volumi e gli snapshot Amazon EBS.
- Rilascia tutti gli Indirizzi IP elastici.
- Annulla la registrazione di tutte le Amazon Machine Image (AMI).
- Termina tutti gli ambienti AWS Elastic Beanstalk.

Se la risorsa è un oggetto in uno spazio di archiviazione Amazon Glacier e se si elimina un archivio prima di aver raggiunto la durata minima di archiviazione, verrà addebitato un costo di eliminazione anticipata proporzionale. La durata minima di archiviazione Amazon Glacier dipende dalla classe di archiviazione utilizzata. Per un riepilogo della durata minima dell'archiviazione per ogni classe di archiviazione, consulta Prestazioni delle classi di archiviazione Amazon S3. Per informazioni sulle modalità di calcolo delle tariffe di eliminazione anticipata, consulta i prezzi di Amazon S3.

Il seguente semplice diagramma di flusso del processo di disattivazione illustra le fasi della disattivazione. Prima di disattivare le risorse, verifica che le risorse identificate per la disattivazione non siano utilizzate dall'organizzazione.



Flusso di disattivazione delle risorse.

Risorse

Documenti correlati:

- AWS Auto Scaling
- AWS Trusted Advisor
- AWS CloudTrail

Video correlati:

- Delete CloudFormation stack but retain some resources
- Find out which user launched Amazon EC2 instance

Esempi correlati:

- Eliminare o interrompere le risorse Amazon EC2
- Scopri quale utente ha lanciato un'istanza Amazon EC2

COST04-BP03 Disattivazione delle risorse

Disattiva le risorse attivate da eventi come audit periodici o modifiche relative all'utilizzo. La disattivazione viene in genere eseguita periodicamente e può essere manuale o automatizzata.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

La frequenza e lo sforzo di ricerca delle risorse inutilizzate dovrebbero riflettere i risparmi potenziali, pertanto un account con costi contenuti deve essere analizzato con una frequenza minore rispetto a un account che ha costi maggiori. Gli eventi di ricerca e disattivazione possono essere avviati da modifiche di stato nel carico di lavoro, ad esempio il termine del ciclo di vita di un prodotto o la sua sostituzione. Le ricerche e gli eventi di disattivazione possono anche essere avviati da eventi esterni, ad esempio cambiamenti nelle condizioni di mercato o cessazione del prodotto.

Passaggi dell'implementazione

- Disattivazione delle risorse: si tratta della fase di ammortamento delle risorse AWS non più necessarie o al termine di un contratto di licenza. Completa tutti i controlli finali prima di passare alla fase di dismissione e disattivazione delle risorse per evitare interruzioni indesiderate durante fasi come l'esecuzione di snapshot o backup. Utilizzando il processo di disattivazione, disattiva tutte le risorse identificate come inutilizzate.

Risorse

Documenti correlati:

- AWS Auto Scaling
- AWS Trusted Advisor

COST04-BP04 Disattivazione automatica delle risorse

Progetta il tuo carico di lavoro in modo da gestire in modo controllato la terminazione delle risorse, identificando e disattivando le risorse non critiche, le risorse non necessarie o quelle a basso utilizzo.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Utilizza l'automazione per ridurre o rimuovere i costi associati al processo di ritiro. Progettare il carico di lavoro per eseguire automaticamente la disattivazione ridurrà i costi complessivi del carico di lavoro durante il suo ciclo di vita. Puoi utilizzare Amazon EC2 Auto Scaling o Application Auto Scaling per eseguire il processo di disattivazione. Puoi anche implementare un codice personalizzato utilizzando un'API o SDK per disattivare automaticamente le risorse del carico di lavoro.

Lo sviluppo delle applicazioni moderne avviene in modalità serverless-first, una strategia che assegnala priorità all'adozione di servizi serverless. AWS ha sviluppato servizi serverless per tutti e tre i livelli dello stack: elaborazione, integrazione e archivi dati. L'utilizzo di un'architettura serverless consente di risparmiare sui costi nei periodi di scarso traffico e di approfittare del dimensionamento automatico.

Passaggi dell'implementazione

- Implementa Amazon EC2 Auto Scaling o Application Auto Scaling: configura le risorse supportate con Amazon EC2 Auto Scaling o Application Auto Scaling. Questi servizi consentono di ottimizzare l'utilizzo e l'efficienza dei costi durante l'utilizzo dei servizi AWS. Quando la domanda diminuisce, questi servizi rimuovono automaticamente la capacità di risorse in eccesso per evitare spese inutili.
- Configura CloudWatch per terminare le istanze: puoi configurare le istanze in modo da terminarle mediante gli allarmi di CloudWatch. Utilizzando i parametri del processo di disattivazione, implementa un allarme con un'operazione Amazon Elastic Compute Cloud. Verifica l'operazione in un ambiente non di produzione prima di eseguire il roll out.
- Implementa il codice all'interno del carico di lavoro: puoi utilizzare l'SDK AWS o AWS CLI per disattivare le risorse del carico di lavoro. Implementa il codice all'interno dell'applicazione che si integra con AWS e termina o rimuove le risorse che non vengono più utilizzate.
- Utilizza servizi serverless: per creare ed eseguire le tue applicazioni, assegna ai la priorità alla creazione di architetture serverless e architetture basate su eventi in AWS. AWS offre diversi servizi tecnologici serverless che forniscono in modo nativo un utilizzo delle risorse ottimizzato in automatico e una disattivazione automatizzata (ridurre orizzontalmente e aumentare orizzontalmente). Con le applicazioni serverless, l'utilizzo delle risorse viene ottimizzato automaticamente e non si paga mai il provisioning in eccesso.

Risorse

Documenti correlati:

- Amazon EC2 Auto Scaling
- Getting Started with Amazon EC2 Auto Scaling
- Application Auto Scaling
- AWS Trusted Advisor
- Serverless in AWS
- Create Alarms to Stop, Terminate, Reboot, or Recover an Instance
- Aggiungere azioni di terminazione agli allarmi Amazon CloudWatch

Esempi correlati:

- Scheduling automatic deletion of AWS CloudFormation stacks

COST04-BP05 Applicare policy di conservazione dei dati

Definisci le policy di conservazione dei dati sulle risorse supportate per gestire l'eliminazione degli oggetti in base ai requisiti della tua organizzazione. Identifica ed elimina risorse non necessarie oppure orfane e oggetti non più richiesti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Usa le policy di conservazione dei dati e del ciclo di vita per ridurre i costi associati al processo di disattivazione e i costi di archiviazione per le risorse identificate. La definizione delle policy di conservazione dei dati e del ciclo di vita per eseguire l'eliminazione e la migrazione di classi di archiviazione automatizzate contribuirà a ridurre i costi di archiviazione generale durante la sua durata. Puoi usare Amazon Data Lifecycle Manager per automatizzare la creazione e l'eliminazione di snapshot Amazon Elastic Block Store e Amazon Machine Image (AMI) supportate da Amazon EBS e usare il Piano intelligente Amazon S3 o una configurazione del ciclo di vita Amazon S3 per gestire il ciclo di vita dei tuoi oggetti Amazon S3. Puoi anche implementare codice personalizzato utilizzando API o SDK così da creare policy del ciclo di vita e regole di policy per gli oggetti da eliminare in automatico.

Passaggi dell'implementazione

- Utilizza Amazon Data Lifecycle Manager: usa le policy del ciclo di vita su Amazon Data Lifecycle Manager per automatizzare l'eliminazione di snapshot Amazon EBS e AMI supportate da Amazon EBS.

- Imposta la configurazione del ciclo di vita su un bucket: usa la configurazione del ciclo di vita di Amazon S3 su un bucket per definire le operazioni che Amazon S3 deve intraprendere durante il ciclo di vita dell'oggetto, oltre all'eliminazione alla fine del ciclo di vita dello stesso, in base ai requisiti aziendali.

Risorse

Documenti correlati:

- AWS Trusted Advisor
- Amazon Data Lifecycle Manager
- Come creare una configurazione del ciclo di vita dei bucket Amazon S3

Video correlati:

- Automate Amazon EBS Snapshots with Amazon Data Lifecycle Manager
- Empty an Amazon S3 bucket using a lifecycle configuration rule

Esempi correlati:

- Empty an Amazon S3 bucket using a lifecycle configuration rule

Risorse convenienti in termini di costo

Questions

- COST 5. In che modo valuti i costi quando selezioni i servizi?
- COST 6. In che modo raggiungi gli obiettivi di costo quando selezioni il tipo, le dimensioni e il numero delle risorse?
- COST 7. In che modo impieghi i modelli di prezzo per ridurre i costi?
- COST 8. In che modo pianifichi i costi per il trasferimento dei dati?

COST 5. In che modo valuti i costi quando selezioni i servizi?

Amazon EC2, Amazon EBS e Amazon S3 sono servizi AWS del blocco predefinito. I servizi gestiti, come Amazon RDS e Amazon DynamoDB, sono servizi AWS di livello superiore o di livello

applicazione. Selezionando i blocchi predefiniti e i servizi gestiti appropriati, è possibile ottimizzare questo carico di lavoro per i costi. Ad esempio, utilizzando i servizi gestiti, puoi ridurre o eliminare gran parte dei costi generali amministrativi e operativi, liberandotene per lavorare su applicazioni e attività correlate al tuo business.

Best practice

- COST05-BP01 Identificare i requisiti organizzativi in termini di costi
- COST05-BP02 Analisi di tutti i componenti del carico di lavoro
- COST05-BP03 Esecuzione di un'analisi accurata di ciascun componente
- COST05-BP04 Selezione di software con licenze convenienti
- COST05-BP05 Selezione dei componenti del carico di lavoro per ottimizzare i costi in linea con le priorità dell'organizzazione
- COST05-BP06 Esecuzione di un'analisi dei costi per diversi valori di utilizzo nel tempo

COST05-BP01 Identificare i requisiti organizzativi in termini di costi

Lavora con i membri del team per definire il bilanciamento tra l'ottimizzazione dei costi e altri pilastri, come le prestazioni e l'affidabilità, per questo carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Nella maggior parte delle organizzazioni, il reparto di tecnologia dell'informazione (IT) è composto da diversi team di piccole dimensioni, ciascuno con una propria agenda e area di interesse, che riflettono le specializzazioni e le competenze dei suoi membri. È necessario comprendere gli obiettivi, le priorità e le finalità generali dell'organizzazione e in che modo ogni reparto o progetto contribuisce a tali obiettivi. La catalogazione di tutte le risorse essenziali, inclusi personale, attrezzature, tecnologia, materiali e servizi esterni, è fondamentale per il raggiungimento degli obiettivi organizzativi e una pianificazione precisa del budget. L'adozione di questo approccio sistematico all'identificazione e alla comprensione dei costi è fondamentale per stabilire un piano dei costi realistico e affidabile per l'organizzazione.

Al momento di selezionare i servizi per un carico di lavoro, è fondamentale comprendere le priorità dell'organizzazione. Crea un equilibrio tra l'ottimizzazione dei costi e altri pilastri del AWS Well-Architected Framework, come prestazioni e affidabilità. È necessario eseguire questo processo sistematicamente e regolarmente in modo da acquisire i cambiamenti a livello di obiettivi, condizioni di mercato e dinamiche operative dell'organizzazione. Un carico di lavoro completamente ottimizzato

per i costi è la soluzione più in linea con i requisiti della tua organizzazione, e non necessariamente quella con il costo più basso. Per raccogliere il maggior numero di informazioni, interpella tutti i team all'interno dell'organizzazione, come i team dedicati ai prodotti, di business, tecnici e finanziari. Valuta l'impatto dei compromessi tra interessi concorrenti o approcci alternativi, per aiutare a prendere decisioni informate quando si stabilisce dove concentrare le attività o scegliere una linea di azione.

Ad esempio, accelerare l'introduzione sul mercato di nuove funzionalità può essere preferibile all'ottimizzazione dei costi. Oppure, è possibile scegliere un database relazionale per i dati non relazionali per semplificare la migrazione di un sistema, anziché migrare a un database ottimizzato per il tuo tipo di dati e aggiornare l'applicazione.

Passaggi dell'implementazione

- Identifica i requisiti dell'organizzazione sui costi: organizza riunioni con i membri dei team della tua organizzazione, tra cui i team di gestione dei prodotti, i team proprietari delle applicazioni, i team operativi e di sviluppo, i team di gestione e finanziari. Dai la priorità ai pilastri Well-Architected per questo carico di lavoro e i relativi componenti. L'output dovrebbe essere un elenco ordinato dei pilastri. Puoi anche aggiungere un fattore di ponderazione a ciascun pilastro per indicare il livello di attenzione aggiuntiva assegnato o quanto è simile il livello di attenzione assegnato a due pilastri.
- Analizza il debito tecnico e documentalo: durante la revisione del carico di lavoro, analizza il debito tecnico. Documenta gli elementi lasciati in sospeso per riesaminare il carico di lavoro in un secondo momento, con l'obiettivo di rifattorizzarlo o riprogettarlo per ottimizzarlo ulteriormente. Alle altre parti interessate è fondamentale comunicare in modo chiaro le scelte di compromesso adottate.

Risorse

Best practice correlate:

- REL11-BP07 Progetta il tuo prodotto per soddisfare gli obiettivi di disponibilità e gli accordi sui livelli di servizio di uptime () SLAs
- OPS01-BP06 Valuta i compromessi

Documenti correlati:

- AWS Calcolatore del costo totale di proprietà () TCO
- Classi di archiviazione Amazon S3
- Prodotti cloud

COST05-BP02 Analisi di tutti i componenti del carico di lavoro

Verifica che ogni componente del carico di lavoro venga analizzato, indipendentemente dalle dimensioni attuali o dai costi correnti. L'attività di revisione deve riflettere i potenziali benefici, come i costi correnti e quelli previsti.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

I componenti del carico di lavoro, progettati per fornire valore aziendale all'organizzazione, possono includere vari servizi. Per ogni componente, è possibile scegliere servizi Cloud AWS specifici per soddisfare specifiche esigenze aziendali. Questa selezione potrebbe essere influenzata da fattori quali la familiarità o l'esperienza precedente nell'uso di questi servizi.

Una volta individuati i requisiti dell'organizzazione, come illustrato in COST05-BP01 Identificazione dei requisiti dell'organizzazione sui costi, esegui un'analisi approfondita di tutti i componenti del carico di lavoro. Analizza ogni componente considerando i costi e le dimensioni attuali e previsti. Considera il costo dell'analisi rispetto a qualsiasi potenziale risparmio del carico di lavoro durante il suo ciclo di vita. L'impegno dedicato all'analisi di tutti i componenti di questo carico di lavoro deve corrispondere al potenziale risparmio o ai miglioramenti previsti derivanti dall'ottimizzazione del componente specifico. Ad esempio, se il costo della risorsa proposta è di 10 USD al mese e secondo le previsioni i carichi non dovrebbero superare i 15 USD al mese, spendere un giorno di lavoro per ridurre i costi del 50% (5 USD al mese) potrebbe eccedere il potenziale beneficio nel corso della vita del sistema. Usa una stima basata sui dati, più rapida ed efficiente, per generare il migliore risultato complessivo per questo componente.

I carichi di lavoro possono cambiare nel corso del tempo e il giusto set di servizi potrebbe non essere ottimale se l'architettura o l'utilizzo del carico di lavoro cambiano. L'analisi per la selezione dei servizi deve integrare gli stati del carico di lavoro e i livelli di utilizzo attuali e futuri. Implementare un servizio in funzione dello stato o dell'utilizzo futuro del carico di lavoro può ridurre i costi complessivi, diminuendo o rimuovendo l'impegno necessario per apportare modifiche future. Ad esempio, EMR Serverless potrebbe inizialmente essere la scelta appropriata. Tuttavia, con l'aumento del consumo del servizio, il passaggio a EMR su EC2 potrebbe ridurre i costi per il componente specifico del carico di lavoro.

AWS Cost Explorer e AWS Cost and Usage Report (CUR) possono analizzare i costi di un proof of concept (PoC) o di un ambiente in esecuzione. Puoi anche utilizzare Calcolatore dei prezzi AWS per stimare i costi del carico di lavoro.

Scrivi un flusso di lavoro che dovrà essere usato dai team tecnici per esaminare i carichi di lavoro. Il flusso di lavoro deve essere semplice, ma coprire tutti i passaggi necessari affinché i team comprendano ogni componente del carico di lavoro e i relativi prezzi. L'organizzazione può quindi seguire e personalizzare il flusso di lavoro in base alle esigenze specifiche di ogni team.

1. Elenca ogni servizio in uso per il tuo carico di lavoro: questa pratica è un buon punto di partenza. Identifica tutti i servizi attualmente in uso e da dove derivano i costi.
2. Scopri come funzionano i prezzi per questi servizi: analizza il modello di prezzo di ciascun servizio. Servizi AWS diversi hanno modelli di prezzi diversi in base a fattori come il volume di utilizzo, il trasferimento dei dati e i prezzi specifici delle funzionalità.
3. Concentrati sui servizi che comportano costi di carico di lavoro imprevisti e non in linea con l'utilizzo previsto e il risultato aziendale: individua i valori anomali o i servizi in cui il costo non è proporzionale al valore o all'utilizzo con AWS Cost Explorer o AWS Cost and Usage Report. È importante correlare i costi ai risultati aziendali per poter definire le priorità delle attività di ottimizzazione.
4. Usa AWS Cost Explorer, CloudWatch Logs, log di flusso VPC e Amazon S3 Storage Lens per analizzare la causa principale dei costi elevati: questi strumenti sono fondamentali nella diagnosi dei costi elevati. Ogni servizio offre una visione diversa per osservare e analizzare l'utilizzo e i costi. Ad esempio, Cost Explorer aiuta a determinare le tendenze generali dei costi, CloudWatch Logs fornisce approfondimenti operativi, i log di flusso VPC mostrano il traffico IP e Amazon S3 Storage Lens è utile per l'analisi dell'archiviazione.
5. Utilizza Budget AWS per fissare budget relativi a determinati importi per servizi o account: fissare budget è una soluzione proattiva per la gestione dei costi. Utilizza Budget AWS per definire soglie di budget personalizzate e ricevere avvisi quando i costi superano tali soglie.
6. Configura gli allarmi Amazon CloudWatch per inviare avvisi di fatturazione e utilizzo: configura monitoraggio e avvisi in relazione ai parametri di costo e utilizzo. Gli allarmi CloudWatch possono avvisarti in caso di raggiungimento di determinate soglie, così da migliorare i tempi di risposta dell'intervento.

Favorisci notevoli miglioramenti e risparmi finanziari nel tempo con la revisione strategica di tutti i componenti del carico di lavoro, indipendentemente dalle caratteristiche attuali. L'impegno profuso in questo processo di revisione deve essere ponderato, con un'attenta considerazione dei potenziali vantaggi che si possono ottenere.

Passaggi dell'implementazione

- Elenca i componenti del carico di lavoro: crea un elenco dei componenti del carico di lavoro. Usa questo elenco per verificare che ogni componente sia stato analizzato. Gli impegni sostenuti devono riflettere la criticità del carico di lavoro secondo quanto definito dalle priorità dell'organizzazione. Raggruppa le risorse in modo funzionale, ad esempio in base all'archiviazione del database di produzione, per migliorare l'efficienza se sono presenti più database.
- Assegna priorità all'elenco dei componenti: assegna ai componenti nell'elenco una priorità in termini di impegno richiesto. Tale assegnazione viene in genere eseguita in ordine dal componente più costoso a quello meno costoso o in base alla criticità definita dalle priorità dell'organizzazione.
- Esegui l'analisi: per ciascun componente dell'elenco, esamina le opzioni e i servizi disponibili e scegli l'opzione che si allinea meglio alle priorità dell'organizzazione.

Risorse

Documenti correlati:

- Calcolatore dei prezzi AWS
- AWS Cost Explorer
- Classi di archiviazione Amazon S3
- Prodotti Cloud AWS

Video correlati:

- AWS Cost Optimization Series: CloudWatch

COST05-BP03 Esecuzione di un'analisi accurata di ciascun componente

Considera il costo complessivo per l'organizzazione di ogni componente. Considera il costo totale di proprietà tenendo conto dei costi operativi e di gestione, soprattutto quando si utilizzano i servizi gestiti del provider cloud. L'attività di revisione deve riflettere i potenziali benefici (ad esempio, il tempo speso per l'analisi dovrebbe essere proporzionale al costo dei componenti).

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all’implementazione

Considera il tempo risparmiato, che consentirà al proprio team di concentrarsi sul ritirare il debito tecnico, sull'innovazione e sulle funzionalità che offrono un valore aggiunto e sullo sviluppo di ciò che diversifica il business. Ad esempio, si potrebbe avere la necessità di eseguire il rehosting (lift and shift) del proprio database dall'ambiente on-premises nel cloud il più rapidamente possibile ed eseguire l'ottimizzazione in un secondo momento. Vale la pena soffermarsi sul risparmio possibile che puoi ottenere usando i servizi gestiti su AWS che rimuovono o riducono i costi di licenza. I servizi gestiti su AWS eliminano l'onere operativo e amministrativo legato alla manutenzione di un servizio, come l'applicazione di patch o l'aggiornamento del sistema operativo, consentendoti di concentrarti sull'innovazione e sul business.

Dato che i servizi gestiti operano su scala cloud, possono offrire un costo inferiore per transazione o servizio. Questo vuol dire fare alcune ottimizzazioni potenziali in modo da ottenere benefici tangibili, senza modificare l'architettura principale dell'applicazione. Ad esempio, potresti voler ridurre il tempo dedicato alla gestione delle istanze di database mediante la migrazione a una piattaforma database-as-a-service come Amazon Relational Database Service (Amazon RDS) o migrando la tua applicazione su una piattaforma completamente gestita come AWS Elastic Beanstalk.

Solitamente, i servizi gestiti presentano attributi che si possono impostare per garantire la capacità necessaria. Devi impostare e monitorare questi attributi in modo che la tua capacità in eccesso sia mantenuta al minimo e le prestazioni siano massimizzate. Puoi modificare gli attributi di AWS Managed Services utilizzando la Console di gestione AWS o le API e gli SDK AWS per allineare le risorse necessarie con le variazioni della domanda. Ad esempio, puoi aumentare o diminuire il numero di nodi su un cluster Amazon EMR o Amazon RedShift per aumentare orizzontalmente o ridurre orizzontalmente.

Puoi anche unire più istanze in una risorsa AWS per attivare una densità di utilizzo più elevata. Ad esempio, puoi predisporre più database di dimensioni ridotte su una singola istanza database di Amazon Relational Database Service (Amazon RDS). Quando l'utilizzo si intensifica, puoi migrare uno dei database su un'istanza database Amazon RDS dedicata utilizzando un processo di generazione dello snapshot e ripristino.

Quando predisponi carichi di lavoro su servizi gestiti, devi comprendere i requisiti inerenti alla modifica della capacità del servizio. Tali requisiti solitamente riguardano il tempo, l'impegno e qualunque impatto sul normale funzionamento del carico di lavoro. La risorsa allocata deve offrire il tempo necessario per l'applicazione delle modifiche, pertanto procurati i mezzi necessari a tal fine. L'impegno costante richiesto per modificare i servizi può essere ridotto praticamente a zero grazie

alle API e agli SDK integrati nel sistema, nonché grazie a strumenti di monitoraggio come Amazon CloudWatch.

Amazon RDS, Amazon Redshift e Amazon ElastiCache forniscono un servizio di database gestito, mentre Amazon Athena, Amazon EMR e il Servizio OpenSearch di Amazon forniscono un servizio di analisi gestito.

AMS è un servizio che gestisce l'infrastruttura AWS per conto di clienti e partner aziendali. Fornisce un ambiente sicuro e conforme in cui è possibile distribuire i carichi di lavoro. AMS utilizza modelli operativi cloud aziendali dotati di automazione per consentirti di soddisfare i requisiti aziendali, di passare più rapidamente al cloud e di ridurre i costi di gestione correnti.

Passaggi dell'implementazione

- Esegui un'analisi completa: utilizzando l'elenco dei componenti, analizza ogni componente dalla priorità più alta alla priorità più bassa. Per la priorità più alta e i componenti più costosi, esegui analisi aggiuntive e valuta tutte le opzioni disponibili e il loro impatto a lungo termine. Per i componenti con priorità più bassa, valuta se le modifiche relative all'utilizzo hanno un impatto sulla priorità del componente, quindi esegui un'analisi dello sforzo appropriato.
- Confronta risorse gestite e non gestite: prendi in considerazione i costi operativi delle risorse che gestisci e confrontali con le risorse AWS gestite. Ad esempio, rivedi i tuoi database in esecuzione su istanze Amazon EC2 e confrontali con le opzioni Amazon RDS (un servizio gestito AWS) o Amazon EMR paragonato all'esecuzione di Apache Spark su Amazon EC2. Quando si passa da un carico di lavoro autogestito a un carico di lavoro AWS completamente gestito, esamina attentamente le tue opzioni. I tre fattori più importanti da prendere in considerazione sono il tipo di servizio gestito da utilizzare, il processo che utilizzerai per la migrazione dei dati e la conoscenza del modello di responsabilità condivisa AWS.

Risorse

Documenti correlati:

- AWS Calcolatore del costo totale di proprietà (TCO) di
- Classi di archiviazione Amazon S3
- Prodotti Cloud AWS
- Modello di responsabilità condivisa AWS

Video correlati:

- Why move to a managed database?
- What is Amazon EMR and how can I use it for processing data?

Esempi correlati:

- Perché passare a un database gestito
- Consolidate data from identical SQL Server databases into a single Amazon RDS for SQL Server database using AWS DMS
- Deliver data at scale to Amazon Managed Streaming for Apache Kafka (Amazon MSK)
- Migrate an ASP.NET web application to AWS Elastic Beanstalk

COST05-BP04 Selezione di software con licenze convenienti

Il software open source elimina i costi di licenza del software, che contribuiscono in modo significativo ai costi dei carichi di lavoro. Nei casi in cui il software con licenza sia obbligatorio, evita le licenze legate ad attributi arbitrari, ad esempio CPU, e cerca le licenze legate all'output o ai risultati. Il costo di queste licenze si ridimensiona in base ai vantaggi che offrono.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Il concetto di open source è nato nel contesto dello sviluppo del software per indicare che il software è conforme a determinati criteri di distribuzione gratuita. Il software open source è composto da codice sorgente che chiunque può analizzare, modificare e migliorare. In base ai requisiti aziendali, alle competenze professionali, all'utilizzo previsto o ad altre dipendenze tecnologiche, le organizzazioni possono prendere in considerazione l'utilizzo di software open source in AWS per ridurre al minimo i costi di licenza. In altri termini, utilizzando software open source è possibile ridurre il costo delle licenze software. Con lo scalare del carico di lavoro, l'impatto sui costi può essere significativo.

Misura i vantaggi di usare software con licenza in rapporto ai costi totali per ottimizzare il carico di lavoro. Crea modelli per le eventuali modifiche alla licenza e il relativo impatto sui costi del carico di lavoro. Se un fornitore modifica il costo della licenza del database, valuta come questo incide sull'efficienza complessiva del carico di lavoro. Effettua un'analisi dello storico dei prezzi dei tuoi fornitori per scoprire le tendenze dei cambiamenti relativi alle licenze dei loro prodotti. I costi delle licenze possono scalare indipendentemente dal throughput o dall'utilizzo, come nel caso delle licenze

che si adattano in base all'hardware (licenze legate alla CPU). È necessario evitare queste licenze poiché i costi possono aumentare rapidamente senza che vi siano vantaggi correlati.

Ad esempio, l'utilizzo di un'istanza Amazon EC2 in us-east-1 con un sistema operativo Linux consente di ridurre i costi di circa il 45% rispetto all'esecuzione di un'altra istanza Amazon EC2 eseguita su Windows.

Il Calcolatore dei prezzi AWS offre una soluzione completa per confrontare i costi di varie risorse con diverse opzioni di licenza, come istanze Amazon RDS e diversi motori di database. Inoltre, AWS Cost Explorer fornisce un punto di vista impareggiabile per i costi dei carichi di lavoro esistenti, in particolare quelli derivanti da licenze diverse. Per la gestione delle licenze, AWS License Manager offre una soluzione semplificata per supervisionare e gestire le licenze software. I clienti possono implementare e rendere operativo il loro software open source preferito nel Cloud AWS.

Passaggi dell'implementazione

- Analizza le opzioni di licenza: esamina i termini di licenza del software disponibile. Cerca le versioni open source che dispongono delle funzionalità necessarie e considera se i vantaggi del software con licenza superano i costi. Condizioni convenienti possono rendere il costo del software proporzionato ai vantaggi che offre.
- Analizza i fornitori del software: esamina tutte le modifiche ai prezzi o alle licenze effettuate dal fornitore. Identifica eventuali modifiche non allineate ai risultati, ad esempio termini punitivi per l'esecuzione su hardware o piattaforme di fornitori specifici. Inoltre, verifica il modo in cui vengono eseguiti gli audit e le sanzioni in cui potresti incorrere.

Risorse

Documenti correlati:

- Open Source at AWS
- Calcolatore del costo totale di proprietà (TCO) di AWS
- Classi di archiviazione Amazon S
- Prodotti cloud

Esempi correlati:

- Blog sull'open source

- Blog AWS sull'open source
- Optimization and Licensing Assessment

COST05-BP05 Selezione dei componenti del carico di lavoro per ottimizzare i costi in linea con le priorità dell'organizzazione

Tieni in considerazione il costo nella selezione di tutti i componenti del tuo carico di lavoro. Ciò include l'utilizzo di servizi a livello di applicazione e servizi gestiti o serverless, container o un'architettura basata sugli eventi per ridurre i costi complessivi. Riduci al minimo i costi di licenza utilizzando software open source, software che non hanno costi di licenza o altre alternative per contenere la spesa.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Quando si selezionano tutti i componenti, è necessario considerare il costo dei servizi e delle opzioni. Ciò include l'utilizzo di servizi gestiti e a livello di applicazione, come Amazon Relational Database Service (Amazon RDS), Amazon DynamoDB, Amazon Simple Notification Service (Amazon SNS) e Amazon Simple Email Service (Amazon SES) per ridurre i costi complessivi dell'organizzazione.

Utilizza funzioni serverless e container per il calcolo, come AWS Lambda e Amazon Simple Storage Service (Amazon S3) per i siti Web statici. Se possibile, containerizza la tua applicazione e utilizza servizi di container gestiti di AWS come Amazon Elastic Container Service (Amazon ECS) o Amazon Elastic Kubernetes Service (Amazon EKS).

Riduci al minimo i costi di licenza utilizzando software open source o software che non prevedono tariffe di licenza: ad esempio, Amazon Linux per carichi di lavoro di calcolo oppure esegui la migrazione dei database ad Amazon Aurora.

Puoi utilizzare servizi serverless o a livello di applicazione, come Lambda, Amazon Simple Queue Service (Amazon SQS), Amazon SNS e Amazon SES. Questi servizi eliminano la necessità di gestire una risorsa e forniscono funzioni di esecuzione del codice, servizi di accodamento e consegna dei messaggi. L'altro vantaggio consiste nel ridurre orizzontalmente le prestazioni e i costi in base all'utilizzo, garantendo l'allocazione e l'attribuzione dei costi in modo efficiente.

L'utilizzo dell'architettura basata sugli eventi è inoltre possibile con i servizi serverless. Le architetture basate su eventi funzionano su base push, per cui tutto succede on demand quando l'evento si presenta sul router. In questo modo non devi sostenere i costi di un continuo polling per verificare un

evento. Ciò significa minor consumo di larghezza di banda della rete, minor utilizzo della CPU, minor capacità di parco istanze inattiva e minor numero di handshake SSL/TLS.

Per ulteriori informazioni sulle funzioni serverless, consulta il whitepaper Well-Architected Serverless Application lens.

Passaggi dell'implementazione

- Seleziona ciascun servizio per ottimizzare i costi: utilizzando l'elenco e l'analisi prioritari, seleziona ciascuna opzione che fornisce la migliore corrispondenza con le priorità dell'organizzazione. Invece di aumentare la capacità per soddisfare la domanda, prendi in considerazione altre opzioni che potrebbero offrirti performance migliori a costi inferiori. Ad esempio, è necessario rivedere il traffico previsto per i database su AWS e prendere in considerazione la possibilità di aumentare le dimensioni dell'istanza o di utilizzare servizi Amazon ElastiCache (Redis o Memcached) per fornire meccanismi di cache per i database.
- Valuta l'architettura basata sugli eventi: l'utilizzo dell'architettura serverless consente inoltre di costruire un'architettura basata sugli eventi per applicazioni distribuite basate su microservizi, che aiuta a costruire soluzioni scalabili, resilienti, agili ed economiche.

Risorse

Documenti correlati:

- Calcolatore del costo totale di proprietà (TCO) di AWS
- Serverless in AWS
- Che cos'è un'architettura basata sugli eventi (EDA)?
- Classi di archiviazione Amazon S3
- Prodotti cloud
- Amazon ElastiCache (Redis OSS)

Esempi correlati:

- Getting started with event-driven architecture
- Architettura basata su eventi
- How Statsig runs 100x more cost-effectively using Amazon ElastiCache (Redis OSS)
- Best practices for working with AWS Lambda functions

COST05-BP06 Esecuzione di un'analisi dei costi per diversi valori di utilizzo nel tempo

I carichi di lavoro possono cambiare nel corso del tempo. Alcuni servizi o funzionalità sono più convenienti a diversi livelli di utilizzo. Eseguendo l'analisi su ogni componente nel tempo e in base all'utilizzo previsto, il carico di lavoro rimane conveniente per tutta la sua durata.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Quando AWS rilascia nuovi servizi e funzionalità, è possibile che i servizi ottimali per il carico di lavoro cambino. Tale cambiamento comporta un impegno, che dovrebbe essere commensurato ai vantaggi potenziali. La frequenza di revisione del carico di lavoro dipende dai requisiti dell'organizzazione. Se si tratta di un carico di lavoro con costi importanti, una rapida implementazione dei nuovi servizi massimizzerà i risparmi sui costi. In tal caso una revisione più frequente può risultare vantaggiosa. Un altro stimolo importante per la revisione è il cambiamento dei modelli di utilizzo. Se si verificassero notevoli cambiamenti nell'utilizzo, ciò potrebbe indicare un maggiore vantaggio dei servizi alternativi.

Se si ha bisogno di trasferire i dati nel Cloud AWS è possibile scegliere i numerosi servizi offerti da AWS e gli strumenti dei partner per avere un supporto nella migrazione dei tuoi set di dati, sia che si tratti di file, database, immagini di macchine, volumi a blocchi o persino backup su nastro. Ad esempio, per spostare grandi quantità di dati da e verso AWS o elaborare dati in posizioni edge, è possibile usare uno dei dispositivi AWS dedicati per migrare, in modo contenuto nei costi, petabyte di dati offline. Un altro esempio: per velocità di trasferimento dei dati più elevate, un servizio di connessione diretta può risultare più economico di una VPN e garantire la connettività coerente richiesta per la tua attività.

In base all'analisi dei costi per usi diversi nel tempo, rivedi le tue attività di dimensionamento. Analizza i risultati per vedere se la policy di dimensionamento può essere ottimizzata per aggiungere istanze con tipi di istanze e opzioni di acquisto diversi. Esamina le tue impostazioni per vedere se il minimo può essere ridotto per soddisfare le richieste degli utenti, ma con una dimensione inferiore del parco istanze, e aggiungi più risorse per i momenti attesi di incremento della domanda.

Esegui analisi dei costi per i vari utilizzi nel tempo discutendo con le parti interessate della tua organizzazione e utilizza la funzionalità di previsione di AWS Cost Explorer per anticipare il potenziale impatto delle modifiche ai servizi. Monitora gli avvii dei livelli di utilizzo con Budget AWS, gli allarmi di fatturazione di CloudWatch e AWS Cost Anomaly Detection per identificare e implementare in tempi rapidi i servizi più contenuti nei costi.

Passaggi dell'implementazione

- Definisci modelli di utilizzo previsti: collaborando con la tua organizzazione, ad esempio con i proprietari di prodotti e marketing, documenta quali sono i modelli di utilizzo previsti per il carico di lavoro. Discuti con le parti interessate dell'azienda dell'aumento dell'utilizzo e dei costi storici e previsti e verifica che tali incrementi siano in linea con i requisiti aziendali. Identifica i giorni, le settimane o i mesi di calendario in cui prevedi che un maggior numero di utenti userà le tue risorse AWS, il che indica che dovrai aumentare la capacità delle risorse esistenti o adottare servizi aggiuntivi per ridurre i costi e migliorare le performance.
- Esegui l'analisi dei costi in base all'utilizzo previsto: esegui l'analisi in ciascuno di questi punti mediante i modelli di utilizzo definiti. Lo sforzo di analisi dovrebbe riflettere il potenziale risultato. Ad esempio, se la variazione dell'utilizzo è elevata, è necessario eseguire un'analisi accurata per verificare eventuali costi e modifiche. In altre parole, quando il costo aumenta dovrebbe aumentare anche l'utilizzo per l'azienda.

Risorse

Documenti correlati:

- Calcolatore del costo totale di proprietà (TCO) di AWS
- Classi di archiviazione Amazon S3
- Prodotti cloud
- Amazon EC2 Auto Scaling
- Migrazione dei dati nel cloud
- AWS Snow Family

Video correlati:

- AWS OpsHub for Snow Family

COST 6. In che modo raggiungi gli obiettivi di costo quando selezioni il tipo, le dimensioni e il numero delle risorse?

Assicurati di scegliere la dimensione e il numero delle risorse appropriati per l'attività in questione. Selezionando il tipo, le dimensioni e il numero più convenienti, riduci al minimo gli sprechi.

Best practice

- COST06-BP01 Esecuzione della modellazione dei costi
- COST06-BP02 Selezione di tipo, dimensione e numero di risorse sulla base dei dati
- COST06-BP03 Selezione automatica del tipo e della dimensione della risorsa in base ai parametri
- COST06-BP04 Valutazione dell'utilizzo delle risorse condivise

COST06-BP01 Esecuzione della modellazione dei costi

Identifica i requisiti dell'organizzazione (come le esigenze aziendali e gli impegni esistenti) ed esegui la modellazione dei costi (costi complessivi) del carico di lavoro e di ciascuno dei suoi componenti. Esegui benchmark per il carico di lavoro in base ai diversi carichi previsti e confronta i costi. L'impegno di modellazione deve riflettere il potenziale risultato. Ad esempio, il tempo speso è proporzionale al costo dei componenti.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Esegui la modellazione dei costi per il tuo carico di lavoro e ciascuno dei suoi componenti per stabilire il giusto equilibrio tra le risorse e trova la dimensione appropriata per ciascuna risorsa nel carico di lavoro, sulla base di un determinato livello di prestazioni. La comprensione delle considerazioni sui costi può contribuire a business case dell'organizzazione e processo decisionale quando si valutano i risultati di realizzazione del valore per l'implementazione del carico di lavoro pianificato.

Esegui benchmark per il carico di lavoro in base ai diversi carichi previsti e confronta i costi. L'impegno di modellazione deve riflettere il potenziale risultato. Ad esempio, il tempo speso dovrebbe essere proporzionale al costo dei componenti o ai risparmi previsti. Per le best practice, consulta la sezione Revisione del pilastro dell'efficienza delle prestazioni del Framework AWS Well-Architected.

Ad esempio, per la creazione di modelli dei costi per un carico di lavoro costituito da risorse di calcolo, AWS Compute Optimizer può contribuire alla modellazione dei costi per l'esecuzione dei carichi di lavoro. Fornisce consigli di dimensionamento appropriato per le risorse di calcolo in base a una valutazione cronologica dell'utilizzo. Assicurati che gli agenti CloudWatch siano distribuiti sulle istanze Amazon EC2 per raccogliere le metriche della memoria che aiutano a fornire raccomandazioni più accurate all'interno di AWS Compute Optimizer. Questa è l'origine dati ideale per le risorse di calcolo poiché si tratta di un servizio gratuito e utilizza il machine learning per generare più raccomandazioni a seconda dei livelli di rischio.

Esistono diversi servizi che puoi utilizzare con log personalizzati come origini dati per operazioni di ridimensionamento corretto per altri servizi e componenti del carico di lavoro, come AWS Trusted Advisor, Amazon CloudWatch e Amazon CloudWatch Logs. AWS Trusted Advisor controlla le risorse e contrassegna quelle a scarso utilizzo, il che può essere utile per il dimensionamento ottimale delle risorse e per la creazione di modelli di costo.

Di seguito sono riportate le raccomandazioni relative a parametri e dati di modellazione dei costi:

- Il monitoraggio deve corrispondere in modo preciso all'esperienza degli utenti. Seleziona la granularità corretta per un dato periodo di tempo e scegli in modo ponderato il 99° percentile o quello massimo invece del valore medio.
- Seleziona la granularità corretta per il periodo di analisi richiesto per coprire tutti i cicli del carico di lavoro. Ad esempio, se esegui un'analisi di due settimane, potresti ignorare un ciclo mensile di utilizzo elevato, con conseguente provisioning insufficiente.
- Scegli i servizi AWS giusti per il carico di lavoro pianificato considerando gli impegni esistenti, i modelli di prezzo selezionati per altri carichi di lavoro e la capacità di innovare più rapidamente e di concentrarsi sul valore del core business.

Passaggi dell'implementazione

- Esegui la modellazione dei costi per le risorse: implementa il carico di lavoro o una proof of concept in un account separato con i tipi di risorse e le dimensioni specifiche da testare. Esegui il carico di lavoro con i dati di test e registra i risultati di output, insieme ai dati relativi ai costi per il tempo in cui è stato eseguito il test. Quindi, implementa di nuovo il carico di lavoro o modifica tipi e dimensioni delle risorse ed esegui nuovamente il test. Includi i costi di licenza di qualsiasi prodotto che si possa utilizzare con queste risorse e i costi operativi stimati (manodopera o tecnici) per l'implementazione e la gestione di queste risorse durante la creazione di modelli di costo. Considera la modellazione dei costi per un periodo (orario, giornaliero, mensile, annuale o triennale).

Risorse

Documenti correlati:

- AWS Auto Scaling
- Identificare le opportunità per il ridimensionamento corretto
- Funzionalità di Amazon CloudWatch

- Cost Optimization: Amazon EC2 Right Sizing
- AWS Compute Optimizer
- Calcolatore dei prezzi AWS

Esempi correlati:

- Esegui una modellazione dei costi basata sui dati
- Stima il costo delle configurazioni di risorse AWS pianificate
- Scegli gli strumenti AWS corretti

COST06-BP02 Selezione di tipo, dimensione e numero di risorse sulla base dei dati

Seleziona la dimensione o il tipo di risorsa in base ai dati relativi a carico di lavoro e caratteristiche delle risorse come, ad esempio, calcolo, memoria, throughput o scrittura intensiva. Questa selezione è tipicamente effettuata utilizzando una versione precedente (on-premises) del carico di lavoro, utilizzando la documentazione o altre fonti di informazione sul carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Amazon EC2 offre un'ampia selezione di tipi di istanza con vari livelli di CPU, memoria, archiviazione e capacità di rete per adattarsi a diversi casi d'uso. Questi tipi di istanza offrono diverse combinazioni di CPU, memoria, archiviazione e funzionalità di rete, che garantiscono versatilità nella scelta delle risorse giuste per i tuoi progetti. Ogni tipo di istanza è disponibile in più dimensioni, per consentire di adattare le risorse alle richieste del carico di lavoro. Per determinare il tipo di istanza necessario, acquisisci i dettagli sui requisiti di sistema dell'applicazione o del software che intendi eseguire sull'istanza. Tali dettagli devono includere le informazioni seguenti:

- Sistema operativo
- Numero di core della CPU
- Core della GPU
- Quantità di memoria di sistema (RAM)
- Tipo e spazio di archiviazione
- Requisiti di larghezza di banda della rete

Identifica lo scopo dei requisiti di calcolo e l'istanza necessaria, quindi analizza le varie famiglie di istanze Amazon EC2. Amazon offre le seguenti famiglie di tipi di istanza:

- Uso generico
- Ottimizzata per il calcolo
- Ottimizzata per la memoria
- Ottimizzata per lo storage
- Calcolo accelerato
- Ottimizzate per il calcolo ad alte prestazioni (HPC)

Per ulteriori informazioni circa scopi e casi d'uso specifici che una particolare famiglia di istanze Amazon EC2 può soddisfare, consulta AWS Instance types.

L'acquisizione dei requisiti di sistema è fondamentale per selezionare famiglia e tipo di istanze specifici più adatti alle proprie esigenze. I nomi dei tipi di istanza sono composti dal nome della famiglia e dalla dimensione dell'istanza. Ad esempio, l'istanza t2.micro appartiene alla famiglia T2 ed è di dimensioni ridotte.

Seleziona la dimensione o il tipo di risorsa in base al carico di lavoro e alle caratteristiche delle risorse come, ad esempio, calcolo, memoria, throughput o uso intensivo di operazioni di scrittura. Questa selezione è in genere effettuata ricorrendo alla modellazione dei costi, a una versione precedente del carico di lavoro (ad esempio una versione on-premises), alla documentazione o ad altre fonti di informazione sul carico di lavoro (come whitepaper o soluzioni pubblicate). L'uso di strumenti di gestione dei costi o calcolatori dei prezzi AWS può favorire l'adozione di decisioni informate su tipi, dimensioni e configurazioni delle istanze.

Passaggi dell'implementazione

- Seleziona le risorse in base ai dati: usa i dati sulla modellazione del costo per selezionare il livello di utilizzo del carico di lavoro previsto e scegli il tipo e le dimensioni delle risorse specificate. Basandoti sui dati di modellazione dei costi, determina il numero di CPU virtuali, la memoria totale (GiB), il volume dell'archivio dell'istanza locale (GB), i volumi Amazon EBS e il livello di prestazioni della rete, tenendo conto della velocità di trasferimento dei dati richiesta per l'istanza. Effettua sempre selezioni basate su analisi dettagliate e dati accurati per ottimizzare le prestazioni e contemporaneamente gestire i costi in modo efficace.

Risorse

Documenti correlati:

- AWS Instance types
- AWS Auto Scaling
- Funzionalità di Amazon CloudWatch
- Cost Optimization: EC2 Right Sizing

Video correlati:

- Selecting the right Amazon EC2 instance for your workloads
- Right size your service

Esempi correlati:

- It just got easier to discover and compare Amazon EC2 instance types

COST06-BP03 Selezione automatica del tipo e della dimensione della risorsa in base ai parametri

Utilizza i parametri del carico di lavoro in esecuzione per selezionare la dimensione e il tipo corretti per ottimizzare i costi. Esegui il provisioning in modo corretto di throughput, dimensione e spazio di archiviazione per servizi di calcolo, memorizzazione, gestione dati e di rete. Questa operazione può essere eseguita con un ciclo di feedback, ad esempio il dimensionamento automatico o tramite codice personalizzato nel carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Crea un ciclo di feedback all'interno del carico di lavoro che utilizza i parametri attivi del carico di lavoro in esecuzione per apportarvi modifiche. Puoi utilizzare un servizio gestito, ad esempio AWS Auto Scaling, configurato per eseguire le operazioni di dimensionamento corretto per te. AWS fornisce inoltre API, SDK e funzionalità per modificare le risorse con il minimo sforzo. È possibile programmare un carico di lavoro per arrestare e avviare un'istanza Amazon EC2 per consentire una modifica delle dimensioni dell'istanza o del tipo di istanza. Ciò offre i vantaggi del dimensionamento appropriato, eliminando al contempo quasi tutti i costi operativi necessari per apportare la modifica.

Alcuni servizi AWS hanno integrato la selezione automatica del tipo o della dimensione, ad esempio Amazon Simple Storage Service Intelligent-Tiering. Il Piano intelligente Amazon S3 sposta automaticamente i dati tra due livelli di accesso: frequente e poco frequente, in base ai tuoi modelli di utilizzo.

Passaggi dell'implementazione

- Aumenta l'osservabilità configurando le metriche del carico di lavoro: acquisisci le metriche chiave per il carico di lavoro. Queste metriche, come l'output del carico di lavoro, forniscono indicazioni sull'esperienza del cliente e sulle differenze tra tipi e dimensioni di risorse, come l'utilizzo di CPU e memoria. Per le risorse di calcolo, analizza i dati sulle prestazioni per dimensionare correttamente le istanze Amazon EC2. Identifica le istanze inattive e quelle sottoutilizzate. Le metriche chiave da cercare sono l'utilizzo di CPU e memoria (ad esempio, il 40% di utilizzo della CPU nel 90% delle volte, come illustrato in Rightsizing with AWS Compute Optimizer and Memory Utilization Enabled). Identifica le istanze con un utilizzo massimo della CPU e della memoria inferiore al 40% su un periodo di quattro settimane. Queste sono le istanze in cui occorre dimensionare correttamente il sistema per ridurre i costi. Per le risorse di archiviazione, come Amazon S3, puoi utilizzare Amazon S3 Storage Lens, che per impostazione predefinita ti consente di visualizzare 28 parametri in varie categorie a livello di bucket e 14 giorni di dati storici nei pannelli di controllo. Per analizzare specifici parametri, si possono applicare dei filtri su riepilogo e ottimizzazione dei costi o eventi all'interno del pannello di controllo di Amazon S3 Storage Lens.
- Visualizza i consigli per il ridimensionamento corretto: utilizza i consigli sul ridimensionamento corretto e in AWS Compute Optimizer e lo strumento per il ridimensionamento corretto di Amazon EC2 nella console di gestione dei costi o verifica il ridimensionamento corretto di AWS Trusted Advisor per apportare modifiche al carico di lavoro. È importante utilizzare gli strumenti giusti per il corretto dimensionamento delle varie risorse e seguire le linee guida per il corretto dimensionamento, che si tratti di un'istanza Amazon EC2, classi di archiviazione AWS o di tipi di istanze Amazon RDS. Per le risorse di archiviazione è possibile utilizzare Amazon S3 Storage Lens, che offre visibilità sull'utilizzo dello spazio di archiviazione di oggetti e sulle tendenze delle attività e fornisce raccomandazioni operative per ottimizzare i costi e applicare le best practice di protezione dei dati. Grazie ai consigli contestuali che Amazon S3 Storage Lens estrapola dall'analisi dei parametri all'interno dell'organizzazione, puoi adottare misure immediate per ottimizzare l'archiviazione.
- Seleziona il tipo di risorse ed esegui il dimensionamento in automatico sulla base delle metriche: utilizza i parametri del carico di lavoro per seleziona manualmente o in automatico le risorse del carico di lavoro. Per le risorse di calcolo, la configurazione di AWS Auto Scaling o l'implementazione di codice all'interno dell'applicazione può ridurre lo sforzo necessario in caso

di modifiche frequenti e permettere di implementare potenzialmente eventuali modifiche più velocemente rispetto a un processo manuale. È possibile avviare e scalare automaticamente un parco di istanze on demand e istanze spot all'interno di un singolo gruppo con un singolo gruppo Auto Scaling. Oltre a ricevere sconti per l'utilizzo di Istanze Spot, è possibile utilizzare Istanze riservate o Savings Plan per ricevere tariffe scontate sul normale prezzo delle istanze on demand. Tutti questi fattori consentono di ottimizzare i risparmi sui costi delle istanze Amazon EC2 e di determinare il dimensionamento e le prestazioni desiderate per la tua applicazione. Puoi anche usare una strategia di selezione del tipo di istanza basata sugli attributi (ABS) nei gruppi Auto Scaling (ASG), così da esprimere i requisiti dell'istanza come un set di attributi, ad esempio vCPU, memoria e spazio di archiviazione. È possibile utilizzare automaticamente i tipi di istanza di nuova generazione quando vengono rilasciati e accedere a una gamma più ampia di capacità con le istanze spot di Amazon EC2. Amazon EC2 Fleet e Amazon EC2 Auto Scaling selezionano e avviano istanze che corrispondono agli attributi specificati, eliminando la necessità di scegliere manualmente i tipi di istanza. Per le risorse di archiviazione, puoi utilizzare le funzionalità del Piano intelligente Amazon S3 e di Amazon EFS Infrequent Access, che consentono di selezionare in automatico classi di archiviazione con risparmi automatici sui costi di archiviazione in caso di modifica ai modelli di accesso ai dati, senza influenzare prestazioni o sovraccarico operativo.

Risorse

Documenti correlati:

- AWS Auto Scaling
- AWS Right-Sizing
- AWS Compute Optimizer
- Funzionalità di Amazon CloudWatch
- CloudWatch Getting Set Up
- CloudWatch Publishing Custom Metrics
- Getting Started with Amazon EC2 Auto Scaling
- Amazon S3 Storage Lens
- Amazon S3 Intelligent-Tiering
- Amazon EFS Infrequent Access
- Avvia un'istanza Amazon EC2 utilizzando l'SDK

Video correlati:

- Right Size Your Services

Esempi correlati:

- Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet
- Optimizing Amazon Elastic Container Service for cost using scheduled scaling
- Predictive scaling with Amazon EC2 Auto Scaling
- Optimize Costs and Gain Visibility into Usage with Amazon S3 Storage Lens

COST06-BP04 Valutazione dell'utilizzo delle risorse condivise

Per i servizi già implementati a livello di organizzazione per più business unit, valuta l'uso delle risorse condivise per aumentare l'utilizzo e ridurre il costo totale di proprietà (TCO). L'utilizzo delle risorse condivise può essere un'opzione conveniente per centralizzare gestione e costi mediante le soluzioni esistenti, condividendo i componenti o in entrambi i casi. Gestisci le funzioni comuni, come monitoraggio, backup e connettività, entro il limite dell'account o in un account dedicato. Inoltre, puoi diminuire i costi implementando la standardizzazione, riducendo duplicazione e complessità.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Se più carichi di lavoro eseguono la stessa funzione, utilizza soluzioni esistenti e componenti condivisi per migliorare la gestione e ottimizzare i costi. Prendi in considerazione l'utilizzo delle risorse esistenti, in particolare quelle condivise, come server di database non di produzione o servizi di directory, per contenere i costi del cloud seguendo le best practice di sicurezza e le normative dell'organizzazione. Per realizzare valore ed efficienza ottimali, è fondamentale utilizzare report di showback e meccanismi di chargeback per riallocare i costi alle aree pertinenti dell'azienda che determinano i consumi.

Con showback si fa riferimento ai report che suddividono i costi del cloud in categorie attribuibili, come consumatori, business unit, account di contabilità generale o altre entità responsabili. L'obiettivo dei report di showback è mostrare a team, business unit o singole persone il costo delle risorse cloud consumate.

Per chargeback si intende l'allocazione della spesa per i servizi centrali alle unità di costo in base a una strategia adatta a uno specifico processo di gestione finanziaria. Per i clienti, il chargeback

addebita il costo sostenuto da un account di servizi condivisi a diverse categorie di costi finanziari definite per un processo di report dei clienti. Stabilendo i meccanismi di chargeback, puoi dichiarare i costi sostenuti da diverse business unit, prodotti e team.

È possibile classificare i carichi di lavoro come critici e non critici. Sulla base di tale classificazione, utilizza le risorse condivise con configurazioni generali per i carichi di lavoro meno critici. Per ottimizzare ulteriormente i costi, usa i server dedicati esclusivamente per i carichi di lavoro critici. Condividi o alloca le risorse in più account per gestirle in modo efficiente. La condivisione è sicura e non compromette la struttura organizzativa anche in caso di separazione di ambienti di sviluppo, test e produzione.

Per migliorare la comprensione e ottimizzare i costi e l'utilizzo delle applicazioni containerizzate, utilizza i dati di allocazione dei costi suddivisi che consentono di allocare i costi alle singole entità aziendali in base al modo in cui l'applicazione consuma le risorse di calcolo e memoria condivise. Con i dati di allocazione dei costi suddivisi puoi ottenere uno showback e un chargeback a livello di attività nei carichi di lavoro dei container in esecuzione su Amazon Elastic Container Service (Amazon ECS) o Amazon Elastic Kubernetes Service (Amazon EKS).

Per le architetture distribuite, crea un VPC di servizi condivisi che fornisca l'accesso centralizzato ai servizi condivisi richiesti dai carichi di lavoro in ogni VPC. Questi servizi condivisi possono includere risorse quali servizi di directory o endpoint VPC. Per ridurre spese e costi amministrativi, condividi le risorse da una posizione centrale invece di crearle in ogni VPC.

Quando si utilizzano le risorse condivise, è possibile ridurre i costi operativi, massimizzare l'utilizzo delle risorse e migliorare la coerenza. In una progettazione multi-account, puoi risparmiare sui costi ospitando alcuni servizi AWS a livello centrale, accedendovi tramite diverse applicazioni e account in un hub. Puoi usare AWS Resource Access Manager (AWS RAM) per condividere altre risorse comuni, come sottoreti VPC e collegamenti AWS Transit Gateway, AWS Network Firewall o pipeline IA di Amazon SageMaker. In un ambiente multi-account, usa AWS RAM per creare una risorsa una sola volta e condividerla con altri account.

Le organizzazioni devono applicare i tag in modo efficace ai costi condivisi e verificare che non vi siano parti significative dei costi senza tag o allocazione. Se non si allocano i costi condivisi in modo efficace e nessuno se ne assume la responsabilità della gestione, i costi condivisi del cloud possono aumentare vertiginosamente. È necessario sapere dove sostieni i costi a livello di risorse, carico di lavoro, team oppure organizzazione poiché queste informazioni migliorano la tua comprensione del valore fornito al livello applicabile rispetto ai risultati aziendali raggiunti. In definitiva, le organizzazioni ottengono il vantaggio del risparmio sui costi grazie alla condivisione dell'infrastruttura cloud. Incoraggia l'allocazione dei costi sulle risorse condivise del cloud per ottimizzare la spesa del cloud.

Passaggi dell'implementazione

- Valutazione dele risorse esistenti: esamina i carichi di lavoro esistenti che utilizzano servizi simili per il tuo carico di lavoro. A seconda dei componenti del carico di lavoro, considera le piattaforme esistenti, se la logica aziendale o i requisiti tecnici lo consentono.
- Utilizzo della condivisione delle risorse in AWS RAM e applicazione di restrizioni di conseguenza: utilizza AWS RAM per condividere le risorse con altri account AWS all'interno dell'organizzazione. Con la condivisione non dovrai duplicare le risorse in più account e riduci al minimo l'onere operativo della manutenzione delle risorse. Questo processo ti consente inoltre di condividere in modo sicuro le risorse create con i ruoli e gli utenti del proprio account e di altri Account AWS.
- Tag delle risorse: effettua il tag delle risorse candidate alla rendicontazione dei costi e classificale in categorie di costo. Attiva questi tag delle risorse relativi ai costi per l'allocazione dei costi per ottenere visibilità sull'utilizzo delle risorse AWS. Concentrati sulla creazione di un livello adeguato di granularità rispetto alla visibilità dei costi e dell'utilizzo e influenza i comportamenti di consumo del cloud attraverso la creazione di report sull'allocazione dei costi e il monitoraggio dei KPI.

Risorse

Best practice correlate:

- SEC03-BP08 Condivisione delle risorse in modo sicuro all'interno dell'organizzazione

Documenti correlati:

- What is AWS Resource Access Manager?
- AWS services that you can use with AWS Organizations
- Shareable AWS resources
- AWS Cost and Usage (CUR) Queries

Video correlati:

- AWS Resource Access Manager - granular access control with managed permissions
- How to design your AWS cost allocation strategy
- AWS Cost Categories

Esempi correlati:

- How-to chargeback shared services: An AWS Transit Gateway example
- How to build a chargeback/showback model for Savings Plans using the CUR
- Using VPC Sharing for a Cost-Effective Multi-Account Microservice Architecture
- Improve cost visibility of Amazon EKS with AWS Split Cost Allocation Data
- Improve cost visibility of Amazon ECS and AWS Batch with AWS Split Cost Allocation Data

COST 7. In che modo impieghi i modelli di prezzo per ridurre i costi?

Utilizza il modello di prezzo più appropriato per le tue risorse per ridurre al minimo le spese.

Best practice

- COST07-BP01 Esecuzione di un'analisi del modello di prezzo
- COST07-BP02 Scelta delle regioni in base al costo
- COST07-BP03 Selezione di contratti di terze parti con condizioni economicamente convenienti
- COST07-BP04 Implementazione di modelli di determinazione dei prezzi per tutti i componenti del carico di lavoro
- COST07-BP05 Esecuzione dell'analisi del modello di prezzo a livello di account di gestione

COST07-BP01 Esecuzione di un'analisi del modello di prezzo

Analizza ogni componente del carico di lavoro. Determina se il componente e le risorse saranno in esecuzione per periodi prolungati (per sconti a fronte di impegni) o dinamici e di breve durata (per spot oppure on demand). Esegui un'analisi sul carico di lavoro utilizzando i suggerimenti degli strumenti di gestione dei costi e applica le regole aziendali ai suggerimenti per ottenere rendimenti elevati.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

AWS offre diversi modelli di prezzo che consentono di pagare per le risorse nel modo più conveniente e adatto alle esigenze della tua organizzazione e in base al prodotto. Lavora con i tuoi team per stabilire il modello di prezzi più appropriato. Spesso il modello di prezzi è costituito da più opzioni, in base alla tua disponibilità

Le istanze on demand consentono di pagare per la capacità di calcolo o di database all'ora o al secondo (minimo 60 secondi), in base alle istanze in esecuzione, senza impegni a lungo termine o pagamenti anticipati.

I Savings Plans sono un modello di prezzo flessibile che offre prezzi conveniente per Amazon EC2, Lambda e per l'utilizzo di AWS Fargate, in cambio di un impegno a un utilizzo costante (misurato in dollari all'ora) per un anno o tre anni.

Le istanze spot sono un meccanismo di determinazione dei prezzi di Amazon EC2 con cui richiedere capacità di calcolo di riserva a una tariffa oraria scontata (fino al 90% di sconto sul prezzo on-demand) senza impegno anticipato.

Le istanze riservate offrono uno sconto fino al 75% pagando in anticipo la capacità. Per ulteriori informazioni, consulta Ottimizzazione dei costi con le prenotazioni.

Potresti scegliere di includere un Savings Plans per le risorse associate alla produzione, alla qualità e agli ambienti di sviluppo. In alternativa, poiché le risorse dell'ambiente di sperimentazione (sandbox) vengono attivate solo quando necessario, è possibile scegliere un modello on demand per le risorse di quell'ambiente. Usa le istanze spot di Amazon per ridurre i costi di Amazon EC2 o utilizza Savings Plans per il calcolo per ridurre i costi di Amazon EC2, Fargate e Lambda. Lo strumento per i suggerimenti di AWS Cost Explorer offre sconti sugli impegni con i Saving Plans.

Se hai acquistato istanze riservate per Amazon EC2 in passato o hai stabilito pratiche di allocazione dei costi all'interno della tua organizzazione, puoi continuare a utilizzare le istanze riservate di Amazon EC2 per il momento. Tuttavia, ti consigliamo di lavorare su una strategia per usare i Savings Plans in futuro come meccanismo più flessibile di risparmio sui costi. Puoi aggiornare i suggerimenti dei Savings Plans (SP) in AWS Cost Management per generare nuovi suggerimenti di Savings Plans in qualsiasi momento. Utilizza le istanze riservate (RI) per ridurre i costi di Amazon RDS, Amazon Redshift, Amazon ElastiCache, e del servizio OpenSearch di Amazon Savings Plans e le istanze riservate sono disponibili in tre opzioni: pagamento anticipato totale, pagamento anticipato parziale e nessun pagamento anticipato. Usa le raccomandazioni fornite nei consigli di acquisto RI e SP AWS Cost Explorer.

Per trovare opportunità per i carichi di lavoro Spot, utilizza una visualizzazione oraria dell'utilizzo complessivo e cerca periodi regolari di variazione di utilizzo o di elasticità. Puoi usare le istanze Spot per diverse applicazioni flessibili e con tolleranza ai guasti. Tra gli esempi figurano server Web stateless, endpoint di API, applicazioni di big data e analisi, carichi di lavoro containerizzati, CI/CD e altri carichi di lavoro flessibili.

Analizza se le tue istanze Amazon EC2 e Amazon RDS possono essere disattivate quando non le usi (dopo l'orario di lavoro e nei weekend). In questo modo potrai ridurre i costi di almeno il 70% rispetto al loro utilizzo 24 ore su 24, 7 giorni su 7. Se hai cluster Amazon Redshift che devono essere disponibili solo in orari specifici, puoi metterli in pausa e poi riattivarli. Quando il cluster Amazon Redshift o l'istanza Amazon EC2 e Amazon RDS vengono arrestati, la fattura relativa all'elaborazione si arresta e si applicano solo i costi di archiviazione.

Tieni presente che le prenotazione della capacità on-demand (ODCR) non corrispondono a uno sconto sui prezzi. Le prenotazioni della capacità vengono addebitate alla tariffa on-demand equivalente indipendentemente dal fatto che si stia o meno eseguendo istanze nella capacità riservata. Tali prenotazioni devono essere prese in considerazione quando hai bisogno di offrire capacità sufficiente alle risorse che desideri eseguire. Le ODCR non devono essere considerate un impegno nel lungo termine, poiché possono essere annullate quando non ne hai più bisogno, ma possono anche approfittare degli sconti offerti da Savings Plans o dalle Istanze riservate.

Passaggi dell'implementazione

- Analizza l'elasticità del carico di lavoro: utilizzando la granularità oraria in Cost Explorer o un pannello di controllo personalizzato, analizza l'elasticità del carico di lavoro. Vai alla ricerca di modifiche regolari del numero di istanze in esecuzione. Le istanze in esecuzione per brevi periodi di tempo sono candidate per essere istanze spot o parco istanze spot.
  - Well-Architected Lab: Cost Explorer
  - Well-Architected Labs: visualizzazione dei costi
- Esamina i contratti esistenti sui prezzi: esamina i contratti o gli impegni in essere per le esigenze a lungo termine. Analizza ciò di cui disponi ora e fino a che punto gli impegni presi vengono sfruttati. Sfrutta sconti contrattuali preesistenti o accordi aziendali. Gli accordi aziendali consentono ai clienti di personalizzare i contratti per adattarli alle loro esigenze. Per accordi nel lungo termine, prendi in considerazione gli sconti dei prezzi riservati, le istanze riservate o Savings Plans per il tipo di istanza specifico, la famiglia delle istanze, Regione AWS e le zone di disponibilità.
- Esegui un'analisi degli sconti a fronte di un impegno: con Cost Explorer nel tuo account, consulta i consigli relativi a Savings Plans e istanze riservate. Per verificare di implementare le raccomandazioni corrette con gli sconti e i rischi richiesti, segui i Well-Architected labs.

Risorse

Documenti correlati:

- Accessing Reserved Instance recommendations
- Opzioni di acquisto delle istanze
- AWS Enterprise

Video correlati:

- Save up to 90% and run production workloads on Spot

Esempi correlati:

- Well-Architected Lab: Cost Explorer
- Well-Architected Labs: visualizzazione dei costi
- Well-Architected Lab: modelli di prezzo

COST07-BP02 Scelta delle regioni in base al costo

La determinazione dei prezzi delle risorse può essere diversa in ciascuna regione. Individua le differenze di costo a livello regionale ed esegui la distribuzione solo nelle regioni con costi più elevati per soddisfare i requisiti di latenza, residenza dei dati e sovranità dei dati. La considerazione del costo della regione garantisce il pagamento del prezzo complessivo più basso per questo carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

L'infrastruttura Cloud AWS è globale ed è ospitata in più sedi in tutto il mondo. Inoltre, si basa su Regioni AWS, zone di disponibilità, zone locali, AWS Outposts e zone di lunghezza d'onda. Una regione è una posizione fisica nel mondo e ogni regione è un'area geografica separata in cui AWS ha più zone di disponibilità. Le zone di disponibilità, che sono più sedi isolate all'interno di ogni regione, sono costituite da uno o più data center discreti, ciascuno con alimentazione, rete e connettività ridondanti.

Ogni Regione AWS opera nelle condizioni di mercato locale e il prezzo delle risorse è diverso in ogni regione, ad esempio a causa delle differenze nel costo della terra, della fibra, dell'elettricità e delle tasse. Scegli una regione specifica per gestire un componente o tutta la tua soluzione in modo da eseguirla al minor prezzo possibile a livello globale. Usa il calcolatore AWS per stimare i costi del

carico di lavoro in varie regioni, cercando i servizi per tipo di località (regione, zona di lunghezza d'onda e zona locale) e regione.

Quando progetti le tue soluzioni, una best practice da seguire è quella di cercare di posizionare le risorse di calcolo vicino agli utenti per offrire una latenza inferiore e una forte sovranità dei dati. Seleziona la posizione geografica in base alle esigenze di business, privacy dei dati, performance e requisiti di sicurezza. Per le applicazioni con utenti finali globali, utilizza più sedi.

Utilizza le regioni che offrono prezzi più bassi per i servizi AWS per distribuire i carichi di lavoro se non hai obblighi in materia di privacy dei dati, sicurezza e requisiti aziendali. Ad esempio, se la regione predefinita è Asia Pacifico (Sydney) (`ap-southwest-2`), e se non ci sono restrizioni (privacy dei dati, sicurezza, ad esempio) per l'utilizzo di altre regioni, l'implementazione di istanze Amazon EC2 non critiche (sviluppo e test) nella regione Stati Uniti orientali (Virginia settentrionale) (`us-east-1`) costerà meno.

	Conformità	*Latenza*	*Costo*	*Servizi/Caratteristiche*
Regione 1	✓	15 ms	$$	✓
Regione 2	✓	20 ms	$$$	X
Regione 3	✓	80 ms	$	✓
Regione 4	✓	15 ms	$$	✓
Regione 5	✓	20 ms	$$$	X
Regione 6	✓	15 ms	$	✓
Regione 7	✓	80 ms	$	✓
Regione 8	✓	15 ms	$	X

Tabella a matrice delle caratteristiche della regione

La tabella a matrice precedente mostra che la regione 6 è l'opzione migliore per questo scenario specifico perché la latenza è bassa rispetto ad altre regioni, il servizio è disponibile ed è la regione meno costosa.

Passaggi dell'implementazione

- Rivedi i prezzi della Regione AWS: analizza i costi del carico di lavoro nella regione corrente. A partire dai costi più elevati per servizio e tipo di utilizzo, calcola i costi in altre regioni disponibili. Se il risparmio previsto supera il costo di spostamento del componente o del carico di lavoro, esegui la migrazione alla nuova regione.
- Rivedi i requisiti per implementazioni multi-regione: analizza i requisiti e gli obblighi aziendali (privacy dei dati, sicurezza o prestazioni) per scoprire se ci sono restrizioni che impediscono di utilizzare più regioni. Se non ci sono obblighi che limitano l'utilizzo di una sola regione, allora utilizza più regioni.
- Analizza il trasferimento dei dati necessario: nel selezionare le regioni, valuta i costi di trasferimento dei dati. Mantieni i dati vicino ai clienti e alle risorse. Seleziona le Regioni AWS meno costose in cui confluiscono i dati e che richiedono trasferimenti minimi di dati. In base ai requisiti aziendali relativi al trasferimento dei dati, puoi utilizzare Amazon CloudFront, AWS PrivateLink, AWS Direct Connect, e AWS Virtual Private Network al fine di ridurre i costi di rete, nonché per migliorare prestazioni e sicurezza.

Risorse

Documenti correlati:

- Accessing Reserved Instance recommendations
- Prezzi di Amazon EC2
- Opzioni di acquisto delle istanze
- Tabella delle regioni

Video correlati:

- Save up to 90% and run production workloads on Spot

Esempi correlati:

- Overview of Data Transfer Costs for Common Architectures
- Cost Considerations for Global Deployments
- What to Consider when Selecting a Region for your Workloads

COST07-BP03 Selezione di contratti di terze parti con condizioni economicamente convenienti

I contratti e i termini convenienti assicurano che i costi di questi servizi siano ridimensionati in base ai vantaggi che offrono. Seleziona i contratti e i prezzi che si ridimensionano quando forniscono ulteriori vantaggi alla tua organizzazione.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Sul mercato esistono diversi prodotti che possono aiutarti a gestire i costi negli ambienti cloud. In termini di funzionalità possono presentare alcune differenze che dipendono dalle esigenze del cliente, ad esempio alcuni clienti sono più interessati alla governance o alla visibilità dei costi mentre altri all'ottimizzazione di questi ultimi. Un fattore chiave per rendere più efficaci l'ottimizzazione e la governance dei costi è l'utilizzo dello strumento giusto con le funzionalità necessarie combinato al giusto modello di prezzo. Questi prodotti hanno modelli di prezzo diversi. Alcuni addebitano una determinata percentuale dell'importo fatturato mensilmente, mentre altri addebitano una percentuale dei risparmi realizzati. Idealmente, dovresti pagare solo ciò che hai effettivamente utilizzato.

Quando utilizzi soluzioni o servizi di terze parti nel cloud, è importante che le strutture dei prezzi siano allineate ai risultati desiderati. I prezzi devono essere scalati in base ai risultati e al valore che forniscono. Ad esempio, se utilizzi un software che contempla una percentuale del risparmio che fornisce, più risparmi (come risultato) e maggiore sarà l'importo addebitato. I contratti di licenza in cui paghi di più all'aumentare delle spese potrebbero non essere sempre nel tuo interesse ai fini dell'ottimizzazione dei costi. Tuttavia, se il fornitore offre vantaggi evidenti per tutte le voci incluse in fattura, questa tariffa scalare potrebbe essere giustificata.

Ad esempio, una soluzione che fornisce suggerimenti per Amazon EC2 e addebita una percentuale dell'intera fattura può diventare più dispendiosa se utilizzi altri servizi che non procurano alcun vantaggio. Un altro esempio è un servizio gestito che viene addebitato a una percentuale del costo delle risorse gestite. Una dimensione di istanza più grande potrebbe non richiedere necessariamente un maggiore impegno di gestione, ma potrebbe comportare un addebito superiore. Verifica che queste disposizioni tariffarie dei servizi includano un programma di ottimizzazione dei costi o funzionalità di servizio volte a migliorare l'efficienza.

I clienti potrebbero trovare i prodotti sul mercato più avanzati o più facili da usare. È necessario considerare il costo di questi prodotti e valutare i potenziali risultati di ottimizzazione dei costi a lungo termine.

Passaggi dell'implementazione

- Analizza contratti e termini stabiliti con terze parti: esamina i prezzi indicati nei contratti con terze parti. Esegui la modellazione per diversi livelli di utilizzo e considera i nuovi costi, come il nuovo utilizzo del servizio o aumenti dei servizi attuali a causa della crescita del carico di lavoro. Decidi se i costi aggiuntivi forniscono i vantaggi necessari alla tua azienda.

Risorse

Documenti correlati:

- Accessing Reserved Instance recommendations
- Opzioni di acquisto delle istanze

Video correlati:

- Save up to 90% and run production workloads on Spot

COST07-BP04 Implementazione di modelli di determinazione dei prezzi per tutti i componenti del carico di lavoro

Le risorse in esecuzione in modo permanente devono utilizzare la capacità riservata, ad esempio Savings Plans o istanze riservate. La capacità a breve termine è configurata per usare le istanze spot o il parco istanze spot. Le istanze on demand vengono utilizzate solo per carichi di lavoro a breve termine che non possono essere interrotti e che non durano abbastanza a lungo per la capacità riservata, tra il 25% e il 75% del periodo, a seconda del tipo di risorsa.

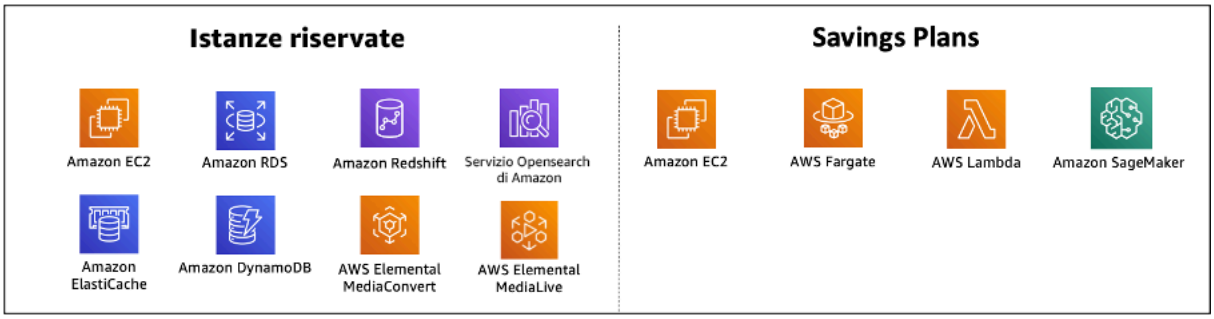
Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all’implementazione

Per migliorare l'efficienza in termini di costi, AWS fornisce diversi consigli sull'impegno economico basati sull'utilizzo pregresso. Puoi utilizzare questi consigli per capire cosa puoi risparmiare e il livello di impegno richiesto. Puoi utilizzare questi servizi come istanze on demand o istanze spot oppure impegnarti per un determinato periodo di tempo e ridurre i costi delle istanze on demand mediante istanze riservate (RI) e Savings Plans (SP). Per ottimizzare il carico di lavoro, è necessario comprendere non solo i singoli componenti del carico di lavoro e i vari servizi AWS, ma anche gli sconti applicati agli impegni, le opzioni di acquisto e le istanze spot per questi servizi.

Considera i requisiti dei componenti del tuo carico di lavoro e valuta i diversi modelli di prezzo per questi servizi. Definisci il requisito di disponibilità dei componenti. Determina l'eventuale presenza di più risorse indipendenti che eseguono la funzione nel carico di lavoro e quali sono i requisiti dello stesso nel corso del tempo. Confronta il costo delle risorse utilizzando il modello di prezzo on demand predefinito e altri modelli applicabili. Tieni conto di qualsiasi potenziale modifica nelle risorse o nei componenti del carico di lavoro.

Analizza, ad esempio, questa architettura di applicazione Web su AWS. Il carico di lavoro di esempio si compone di più servizi AWS, come Amazon Route 53, AWS WAF, Amazon CloudFront, istanze Amazon EC2, istanze Amazon RDS, bilanciatori del carico, storage Amazon S3 e Amazon Elastic File System (Amazon EFS). È necessario esaminare ciascuno di questi servizi e individuare le potenziali opportunità di risparmio sui costi con diversi modelli di prezzo. Alcuni potrebbero essere idonei per le istanze riservate (RI) o per il modello Savings Plans, mentre altri potrebbero essere disponibili solo nelle istanze on demand. Come illustrato nell'immagine seguente, alcuni servizi AWS possono essere eseguiti utilizzando le istanze riservate (RI) o il modello Savings Plans.



Servizi AWS gestiti utilizzando istanze riservate e Savings Plans

Passaggi dell'implementazione

- Implementa modelli di prezzo: partendo dai risultati delle tue analisi, acquista Savings Plans, istanze riservate o implementa istanze spot. Se è il tuo primo acquisto a fronte di impegni, scegli i primi cinque o dieci consigli nell'elenco, quindi monitora e analizza i risultati nel corso del mese successivo o dei due mesi successivi. AWS Cost Management Console ti guida durante l'intero processo. Rivedi i consigli relativi all'istanza riservata (RI) o al modello Savings Plans sulla console, personalizza i consigli (tipo, pagamento e durata) e rivedi l'impegno orario (ad esempio, 20 USD all'ora), quindi aggiungilo al carrello. Gli sconti sono applicati automaticamente all'utilizzo idoneo. Acquista un importo ridotto di sconti a fronte di impegni a cicli regolari, ad esempio ogni 2 settimane o ogni mese. Implementa istanze spot per carichi di lavoro che possono essere interrotti o che sono stateless. Infine, seleziona le istanze Amazon EC2 on demand e alloca le risorse per i requisiti rimanenti.

- Ciclo di revisione del carico di lavoro: implementa un ciclo di revisione per il carico di lavoro che analizzi in modo specifico la copertura del modello di prezzo. Quando il carico di lavoro ha la copertura necessaria, acquista ulteriori sconti a fronte di impegni parzialmente (ogni pochi mesi) o al variare dell'utilizzo dell'organizzazione.

Risorse

Documenti correlati:

- Understanding your Savings Plans recommendations
- Accessing Reserved Instance recommendations
- Modalità di acquisto delle istanze riservate
- Opzioni di acquisto delle istanze
- Spot Instances
- Modelli di prenotazione per altri servizi AWS
- Servizi supportati da Savings Plans

Video correlati:

- Save up to 90% and run production workloads on Spot

Esempi correlati:

- What should you consider before purchasing Savings Plans?
- How can I use Cost Explorer to analyze my spending and usage?

COST07-BP05 Esecuzione dell'analisi del modello di prezzo a livello di account di gestione

Verifica gli strumenti di gestione dei costi e di fatturazione e dai un'occhiata agli sconti suggeriti con impegni e prenotazioni per eseguire analisi regolari a livello di account di gestione.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all’implementazione

L'esecuzione della modellazione dei costi a intervalli regolari garantisce l'implementazione di opportunità di ottimizzazione su più carichi di lavoro. Ad esempio, se più carichi di lavoro utilizzano

istanze on demand a livello aggregato, il rischio di modifica è inferiore e l'implementazione di uno sconto a fronte di impegni permetterà di raggiungere un costo complessivo inferiore. Si consiglia di eseguire l'analisi a cicli regolari a cadenza quindicinale in un mese. In questo modo è possibile effettuare acquisti in piccoli incrementi, così che la copertura dei modelli di prezzo evolva di pari passo con i carichi di lavoro e i relativi componenti.

Usa lo strumento per i suggerimenti AWS Cost Explorer per scoprire opportunità di sconti a fronte di impegni nell'account di gestione. I suggerimenti a livello di account di gestione sono calcolati considerando l'utilizzo di tutti gli account nella tua organizzazione AWS che presenta istanze riservata (RI) o Savings Plans (SP) Vengono inoltre calcolati quando viene attivata la condivisione degli sconti per consigliare un impegno che massimizzi i risparmi su tutti gli account.

Sebbene in molti casi l'acquisto a livello di account di gestione rappresenti un'ottimizzazione che garantisce risparmi massimi, in alcuni casi potresti prendere in considerazione l'acquisto di Savings Plans a livello di account collegato, ad esempio quando desideri che gli sconti si applichino prima all'utilizzo in quel particolare account collegato. I suggerimenti degli account membri sono calcolati a livello di singolo account per massimizzare i risparmi per ogni account isolato. Se il tuo account ha vincoli o impegni sia per istanze riservate (RI) che per Savings Plans (SP), questi verranno applicati nel seguente ordine:

1. RI zonale
2. RI standard
3. RI convertibile
4. Piano di risparmio delle istanze
5. Piano di risparmio di calcolo

Se acquisti un SP a livello di account di gestione, i risparmi verranno applicati in base alla percentuale di sconto dalla più alta alla più bassa. I Savings Plans a livello di account di gestione esaminano tutti gli account collegati e applicano i risparmi ovunque lo sconto sia il più elevato. Se desideri limitare il luogo in cui vengono applicati i risparmi, puoi acquistare un Savings Plan a livello di account collegato e ogni volta che l'account esegue servizi di calcolo idonei, verrà applicato lo sconto. Quando l'account non esegue servizi di calcolo idonei, lo sconto verrà condiviso con gli altri account collegati con lo stesso account di gestione. La condivisione degli sconti è attivata per impostazione predefinita, ma può essere disattivata se necessario.

In una famiglia con fatturazione consolidata, i Savings Plans vengono applicati prima all'utilizzo dell'account del proprietario e, quindi, all'utilizzo degli altri account. Ciò si verifica solo se la

condivisione è abilitata. I tuoi Savings Plans vengono applicati per primi alla percentuale di risparmio più alta. Se ci sono più utilizzi con percentuali di risparmio uguali, i Savings Plans vengono applicati al primo utilizzo con la tariffa dei Savings Plans più bassa. I Savings Plans continuano a essere validi fino all'esaurimento degli usi rimanenti o fino all'esaurimento del tuo impegno. L'eventuale utilizzo residuo viene addebitato in base alle tariffe on demand. Puoi aggiornare i suggerimenti dei Savings Plans in AWS Cost Management per creare nuovi suggerimenti di Savings Plans in qualsiasi momento.

Dopo aver analizzato la flessibilità delle istanze, puoi prendere una decisione in base ai suggerimenti ricevuti. Crea una modellazione dei costi analizzando i costi a breve termine del carico di lavoro rispetto a potenziali diverse opzioni di risorse, analizzando i modelli di prezzo AWS e allineandoli ai requisiti aziendali per scoprire il costo totale di proprietà e le opportunità di ottimizzazione dei costi.

Passaggi dell'implementazione

Esegui un'analisi degli sconti a fronte di un impegno: con Cost Explorer nel tuo account, consulta i consigli su Savings Plans e istanze riservate. Verifica di aver compreso i suggerimenti dei Savings Plans, fai una stima della tua spesa mensile e calcola il risparmio che puoi ottenere su tale intervallo di tempo. Esamina i consigli a livello di account di gestione, calcolati considerando l'utilizzo in tutti gli account membri della tua organizzazione AWS con abilitata la condivisione degli sconti Savings Plans o istanze riservate, per ottenere il massimo risparmio tra gli account. Per assicurarti di implementare le raccomandazioni corrette con gli sconti e i rischi richiesti, segui i Well-Architected Labs.

Risorse

Documenti correlati:

- Come funzionano i prezzi di AWS?
- Opzioni di acquisto delle istanze
- Panoramica del Saving Plan
- Saving Plan recommendations
- Accessing Reserved Instance recommendations
- Understanding your Saving Plans recommendation
- How Savings Plans apply to your AWS usage
- Savings Plans con fatture consolidate

- Attivazione della condivisione di sconti istanze riservate e Savings Plans

Video correlati:

- Save up to 90% and run production workloads on Spot

Esempi correlati:

- Cosa devo considerare prima di acquistare un Savings Plan?
- How can I use rolling Savings Plans to reduce commitment risk?
- Quando usare le istanze Spot

COST 8. In che modo pianifichi i costi per il trasferimento dei dati?

Assicurati di pianificare e monitorare i costi di trasferimento dei dati in modo da poter prendere decisioni sull'architettura per ridurre al minimo i costi. Una modifica piccola ma efficace dell'architettura può ridurre drasticamente i costi operativi nel tempo.

Best practice

- COST08-BP01 Esecuzione della modellazione del trasferimento dei dati
- COST08-BP02 Selezione dei componenti per ottimizzare il costo di trasferimento dei dati
- COST08-BP03 Implementazione dei servizi per ridurre il costo di trasferimento dei dati

COST08-BP01 Esecuzione della modellazione del trasferimento dei dati

Raccogli i requisiti dell'organizzazione ed esegui la modellizzazione del trasferimento dei dati del carico di lavoro e di ciascuno dei suoi componenti. Questo identifica il punto di costo più basso per le sue attuali esigenze di trasferimento dei dati.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Quando si progetta una soluzione nel cloud, i costi del trasferimento dei dati vengono in genere ignorati a causa dell'abitudine di progettare l'architettura utilizzando data center on-premises o per mancanza di conoscenze. I costi del trasferimento dei dati in AWS sono determinati dall'origine, dalla destinazione e dal volume del traffico. Tenere conto di questi costi durante la fase di progettazione

può produrre risparmi. Capire dove avviene il trasferimento dei dati nel carico di lavoro, il costo del trasferimento e i vantaggi associati è molto importante per stimare con precisione il costo totale di proprietà (TCO). In questo modo puoi prendere una decisione consapevole quando si tratta di modificare o accettare una decisione relativa all'architettura. Ad esempio, potresti disporre di una configurazione con più zone di disponibilità dove replichi i dati tra le varie zone di disponibilità.

Puoi modellare i componenti dei servizi che trasferiscono i dati nel carico di lavoro e decidere che si tratta di un costo accettabile (simile a quello del calcolo e dell'archiviazione in entrambe le zone di disponibilità) per ottenere l'affidabilità e la resilienza richieste. Modella i costi in base a livelli differenti di utilizzo. L'utilizzo del carico di lavoro può cambiare nel corso del tempo e servizi differenti possono risultare più convenienti a livelli differenti.

Mentre modelli il trasferimento dei dati, pensa alla quantità di dati acquisiti e alla loro provenienza. Inoltre, considera la quantità di dati elaborati e la capacità di archiviazione o calcolo necessaria. Durante la modellazione, attieniti alle best practice relative alle reti in relazione all'architettura del carico di lavoro per ottimizzare i potenziali costi di trasferimento dei dati.

Calcolatore dei prezzi AWS può aiutarti a vedere i costi stimati per servizi AWS specifici e per il trasferimento di dati previsto. In presenza di un carico di lavoro già in esecuzione (a scopo di test o in un ambiente di preproduzione), utilizza AWS Cost Explorer o AWS Cost and Usage Report (CUR) per analizzare e modellare i costi di trasferimento dei dati. Configura un proof of concept (PoC) o testa il carico di lavoro ed esegui un test con un carico simulato realistico. Puoi modellare i costi in base alle diverse esigenze di carico di lavoro.

Passaggi dell'implementazione

- Identificazione dei requisiti: quali sono l'obiettivo principale e i requisiti aziendali per il trasferimento pianificato dei dati tra origine e destinazione? Qual è il risultato aziendale previsto finale? Acquisisci i requisiti aziendali e definisci il risultato previsto.
- Identificazione di origine e destinazione: quali sono l'origine dati e la relativa destinazione del trasferimento dei dati, ad esempio all'interno delle Regioni AWS, verso i servizi AWS o Internet?
  - Trasferimento di dati all'interno di una Regione AWS
  - Trasferimento di dati tra Regioni AWS
  - Trasferimento di dati verso Internet
- Identificazione delle classificazioni dei dati: qual è la classificazione dei dati per il trasferimento di dati in questione? Di che tipo di dati si tratta? Quali sono le dimensioni dei dati? Con quale frequenza devono essere trasferiti i dati? I dati sono sensibili?

- Identificazione di servizi o strumenti AWS da utilizzare: quali servizi AWSvengono utilizzati per il trasferimento di dati in questione? È possibile utilizzare un servizio già allocato a un altro carico di lavoro?
- Calcolo dei costi di trasferimento dei dati: utilizza i prezzi AWS, nonché il modello di trasferimento dati creato in precedenza, per calcolare i costi di trasferimento dei dati per il carico di lavoro. Calcola i costi di trasferimento dei dati a diversi livelli di utilizzo, ipotizzando incrementi e riduzioni dell'utilizzo del carico di lavoro. Nei casi in cui sono disponibili più opzioni per l'architettura del carico di lavoro valuta i costi di ogni opzione per il confronto.
- Collegamento dei costi ai risultati: per ogni costo di trasferimento dei dati sostenuto, specifica il risultato ottenuto per il carico di lavoro. Se si tratta di un trasferimento tra componenti potrebbe trattarsi di una necessità di disaccoppiamento, se si tratta di un trasferimento tra zone di disponibilità potrebbe trattarsi di una necessità di ridondanza.
- Creazione della modellazione per il trasferimento dei dati: una volta raccolte tutte le informazioni, crea una base concettuale di modellazione del trasferimento dei dati per più casi d'uso e diversi carichi di lavoro.

Risorse

Documenti correlati:

- AWS caching solutions
- Prezzi di AWS
- Prezzi di Amazon EC2
- Prezzi di Amazon VPC
- Understanding data transfer charges

Video correlati:

- Monitoring and Optimizing Your Data Transfer Costs
- S3 Transfer Acceleration

Esempi correlati:

- Overview of Data Transfer Costs for Common Architectures
- AWS Prescriptive Guidance for Networking

COST08-BP02 Selezione dei componenti per ottimizzare il costo di trasferimento dei dati

Tutti i componenti sono selezionati e l'architettura è progettata per ridurre i costi di trasferimento dei dati. Ciò comprende l'utilizzo di componenti come l'ottimizzazione della rete WAN (wide-area network) e le configurazioni con più zona di disponibilità (AZ).

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Una progettazione basata sul trasferimento dei dati riduce i costi del trasferimento stesso. Potrebbe implicare l'uso di reti di distribuzione di contenuti per posizionare i dati vicino agli utenti, oppure l'uso di collegamenti di rete dedicati dalle tue sedi ad AWS. Puoi anche utilizzare l'ottimizzazione WAN e l'ottimizzazione delle applicazioni per ridurre la quantità di dati trasferiti tra i componenti.

Quando si trasferiscono dati verso il Cloud AWS o al suo interno, è essenziale conoscere la destinazione in base a vari casi d'uso, alla natura dei dati e alle risorse di rete disponibili al fine di selezionare i servizi AWS corretti per ottimizzare il trasferimento dei dati. AWS offre una gamma di servizi personalizzati per le diverse esigenze di migrazione dei dati. Seleziona le opzioni di archiviazione di dati e trasferimento di dati opportune in base alle esigenze aziendali all'interno della tua organizzazione.

Quando pianifichi o rivedi l'architettura di un carico di lavoro, considera quanto segue:

- Usa gli endpoint VPC in AWS: con gli endpoint VPC puoi stabilire connessioni private tra VPC e servizi AWS supportati. Ciò consente di evitare l'utilizzo della rete Internet pubblica, che può comportare costi di trasferimento dei dati.
- Usa un gateway NAT: con un gateway NAT, le istanze di una sottorete privata possono connettersi a Internet o servizi esterni al VPC. Verifica se le risorse dietro il gateway NAT che inviano la maggior parte del traffico si trovano nella stessa zona di disponibilità del gateway NAT. In caso negativo, crea nuovi gateway NAT nella stessa zona di disponibilità della risorsa per ridurre i costi di trasferimento dei dati tra zone di disponibilità.
- Usa AWS Direct Connect: Direct Connect bypassa la rete Internet pubblica e stabilisce una connessione diretta e privata tra la rete on-premises e AWS. Ciò può essere più conveniente e coerente rispetto al trasferimento di grandi volumi di dati su Internet.
- Evita il trasferimento di dati tra confini regionali: i trasferimenti di dati tra Regioni AWS (da una regione all'altra) di solito comportano costi. Seguire questo approccio basato sul trasferimento tra regioni dovrebbe essere una decisione molto ponderata. Per ulteriori informazioni, consulta Scenari multi-regione.

- Monitora il trasferimento di dati: con Amazon CloudWatch e i log di flusso VPC, acquisisci dettagli su trasferimento di dati e utilizzo della rete. Analizza le informazioni sul traffico di rete acquisite nei tuoi VPC, come l'indirizzo IP o l'intervallo a livello di interfacce di rete.
- Analizza l'utilizzo della rete: sfrutta strumenti di misurazione e creazione di report come AWS Cost Explorer, i pannelli di controllo CUDOS o CloudWatch per comprendere i costi di trasferimento dei dati del carico di lavoro.

Passaggi dell'implementazione

- Seleziona i componenti per il trasferimento dei dati: utilizzando la modellazione per il trasferimento dei dati illustrata in COST08-BP01 Esecuzione della modellazione del trasferimento dei dati, concentrati su dove si trovano i costi di trasferimento dei dati più elevati o dove sarebbero se l'utilizzo del carico di lavoro cambiasse. Individua architetture alternative o componenti aggiuntivi che eliminano o riducono la necessità di trasferimento dei dati o ne riducono i costi.

Risorse

Best practice correlate:

- COST08-BP01 Esecuzione della modellazione del trasferimento dei dati
- COST08-BP03 Implementazione dei servizi per ridurre il costo di trasferimento dei dati

Documenti correlati:

- Migrazione dei dati nel cloud
- AWS caching solutions
- Deliver content faster with Amazon CloudFront

Esempi correlati:

- Overview of Data Transfer Costs for Common Architectures
- AWS Network Optimization Tips
- Optimize performance and reduce costs for network analytics with VPC Flow Logs in Apache Parquet format

COST08-BP03 Implementazione dei servizi per ridurre il costo di trasferimento dei dati

Implementa i servizi per ridurre il costo di trasferimento dei dati. Ad esempio, utilizza posizioni edge o rete di distribuzione di contenuti (CDN) per fornire contenuti agli utenti finali, crea livelli di memorizzazione nella cache davanti ai database o ai server delle applicazioni e utilizza connessioni di rete dedicate anziché VPN per la connettività al cloud.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Esistono diversi servizi AWS che possono aiutarti a ottimizzare l'utilizzo del trasferimento dei dati di rete. A seconda dei componenti del carico di lavoro, del tipo e dell'architettura cloud, questi servizi possono aiutarti nella compressione, nella memorizzazione nella cache, nella condivisione e distribuzione del traffico sul cloud.

- Amazon CloudFront è una rete globale di distribuzione di contenuti che trasferisce i dati con una latenza ridotta e una velocità di trasferimento elevata. Memorizza nella cache i dati nelle posizioni edge di tutto il mondo, riducendo così il carico sulle tue risorse. Utilizzando CloudFront puoi ridurre l'impegno amministrativo legato alla distribuzione dei contenuti per numeri elevati di utenti a livello globale, con una latenza minima. Il security savings bundle può aiutarti a risparmiare fino al 30% sull'utilizzo di CloudFront se prevedi di aumentarlo nel tempo.
- AWS Direct Connect ti consente di creare una connessione di rete dedicata ad AWS. In questo modo puoi ridurre i costi di rete, aumentare la larghezza di banda e offrire un'esperienza di rete più costante rispetto alle connessioni Internet.
- Site-to-Site VPN consente di stabilire una connessione sicura e privata tra la rete privata e la rete globale AWS. È ideale per piccoli uffici o partner aziendali perché offre una connettività semplificata ed è un servizio completamente gestito ed elastico.
- Gli endpoint VPC consentono la connettività tra i servizi AWS su reti private e possono essere utilizzati per ridurre i costi di trasferimento di dati pubblici e dei gateway NAT. Gli endpoint VPC del gateway non hanno tariffe orarie e supportano Amazon S3 e Amazon DynamoDB. Gli endpoint VPC dell'interfaccia sono forniti da AWS PrivateLink e prevedono una tariffa oraria e un costo di utilizzo per GB.
- I gateway NAT offrono scalabilità e gestione integrate che riducono i costi rispetto a un'istanza NAT autonoma. Per ridurre i costi di trasferimento ed elaborazione dei dati, posiziona i gateway NAT nelle stesse zone di disponibilità delle istanze a elevato traffico e valuta la possibilità di utilizzare gli endpoint VPC per le istanze che devono accedere ad Amazon DynamoDB o Amazon S3

- Utilizza dispositivi AWS Snow Family dotati di risorse di calcolo per raccogliere ed elaborare dati nelle posizioni edge. I dispositivi AWS Snow Family (Snowball Edge, Snowball Edge e Snowmobile) consentono di trasferire petabyte di dati nel Cloud AWS modo conveniente e offline.

Passaggi dell'implementazione

- Implementa i servizi: seleziona i servizi di rete di AWS applicabili in base al servizio e al tipo di carico di lavoro utilizzando la modellazione del trasferimento dei dati e la revisione dei log di flusso VPC. Scopri dove si trovano i costi maggiori e i flussi con volumi più elevati. Esamina i servizi AWS e valuta se esiste un servizio che riduce o rimuove il trasferimento, in particolare nell'ambito delle reti e della distribuzione di contenuti. Individua anche servizi di caching in cui si verifica un accesso ripetuto ai dati o in cui sono presenti grandi quantità di dati.

Risorse

Documenti correlati:

- AWS Direct Connect
- Esplora i prodotti AWS
- AWS caching solutions
- Amazon CloudFront
- AWS Snow Family
- Bundle CloudFront Security Savings Amazon

Video correlati:

- Monitoring and Optimizing Your Data Transfer Costs
- AWS Cost Optimization Series: CloudFront
- How can I reduce data transfer charges for my NAT gateway?

Esempi correlati:

- How-to chargeback shared services: An AWS Transit Gateway example
- Understand AWS data transfer details in depth from cost and usage report using Athena query and QuickSight

- Overview of Data Transfer Costs for Common Architectures
- Using AWS Cost Explorer to analyze data transfer costs
- Cost-Optimizing your AWS architectures by utilizing Amazon CloudFront features
- How can I reduce data transfer charges for my NAT gateway?

Gestione delle risorse di domanda e offerta

Domanda

- COST 9. Come gestisci la domanda e fornisci le risorse?

COST 9. Come gestisci la domanda e fornisci le risorse?

Per avere un carico di lavoro con costo e prestazioni bilanciate, verifica che venga utilizzato tutto ciò per cui paghi ed evita le istanze molto sottoutilizzate. Un parametro di utilizzo distorto, in qualsiasi delle suddette direzioni, ha un impatto negativo sull'organizzazione, sia per i costi operativi (basse prestazioni a causa di un utilizzo eccessivo) che per le spese inerenti a AWS sprecate (a causa di un provisioning eccessivo).

Best practice

- COST09-BP01 Analisi della domanda del carico di lavoro
- COST09-BP02 Implementazione di un buffer o della limitazione (della larghezza di banda della rete) per gestire la domanda
- COST09-BP03 Fornitura dinamica delle risorse

COST09-BP01 Analisi della domanda del carico di lavoro

Analizza la domanda del carico di lavoro nel tempo. Verifica che l'analisi copra l'andamento stagionale e rappresenti accuratamente le condizioni operative per l'intera durata del carico di lavoro. L'attività di analisi deve riflettere i potenziali benefici, ad esempio che il tempo speso sia proporzionale al costo del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

L'analisi della domanda di carichi di lavoro per il cloud computing implica la comprensione dei modelli e delle caratteristiche delle attività di elaborazione avviate nell'ambiente cloud. Questa analisi aiuta gli

utenti a ottimizzare l'allocazione delle risorse, gestire i costi e verificare che le prestazioni soddisfino i livelli richiesti.

Scopri i requisiti del carico di lavoro. I requisiti dell'organizzazione devono indicare i tempi di risposta del carico di lavoro per le richieste. Il tempo di risposta può essere utilizzato per determinare se la domanda è gestita o se l'offerta di risorse cambierà per soddisfare la domanda.

L'analisi deve includere la prevedibilità e la ripetibilità della domanda, la velocità di variazione della domanda e la quantità di variazione della domanda. Esegui l'analisi per un periodo abbastanza lungo da incorporare qualsiasi variazione stagionale, ad esempio l'elaborazione di fine mese o i picchi legati alle festività.

Lo sforzo di analisi dovrebbe riflettere i potenziali vantaggi dell'implementazione della scalabilità. Osserva il costo totale previsto del componente ed eventuali aumenti o riduzioni di utilizzo e costi durante il ciclo di vita del carico di lavoro.

Di seguito sono riportati alcuni aspetti chiave da prendere in considerazione quando si esegue l'analisi della domanda del carico di lavoro per il cloud computing:

1. Utilizzo delle risorse e metriche sulle prestazioni: analizza l'utilizzo nel tempo delle risorse AWS. Determina i modelli di utilizzo di picco e non di picco per ottimizzare l'allocazione delle risorse e le strategie di scalabilità. Monitora le metriche delle prestazioni come tempi di risposta, latenza, throughput e tassi di errore. Queste metriche aiutano a valutare lo stato e l'efficienza complessive dell'infrastruttura cloud.
2. Comportamento in termini di dimensionamento di utenti e applicazioni: analizza il comportamento degli utenti e il relativo impatto sulla domanda del carico di lavoro. L'esame dei modelli di traffico degli utenti aiuta a migliorare la fornitura di contenuti e la reattività delle applicazioni. Analizza la modalità di dimensionamento dei carichi di lavoro in base all'aumento della domanda. Determina se i parametri di dimensionamento automatico sono configurati correttamente ed efficacemente per gestire le fluttuazioni del carico.
3. Tipi di carico di lavoro: identifica i diversi tipi di carichi di lavoro in esecuzione nel cloud, come l'elaborazione in batch, l'elaborazione dei dati in tempo reale, le applicazioni Web, i database o i processi di machine learning. Ogni tipo di carico di lavoro può avere requisiti di risorse e profili di prestazioni diversi.
4. Accordi sul livello di servizio (SLA): confronta le prestazioni effettive con gli SLA per garantire la conformità e identificare le aree che necessitano di miglioramento.

Puoi utilizzare Amazon CloudWatch per raccogliere e monitorare metriche e file di log, impostare allarmi e reagire automaticamente ai cambiamenti nelle risorse AWS. Puoi anche usare Amazon CloudWatch per ottenere visibilità a livello di sistema su utilizzo delle risorse, prestazioni delle applicazioni e stato di integrità operativa.

Con AWS Trusted Advisor, puoi allocare le tue risorse seguendo le best practice per migliorare le prestazioni e l'affidabilità del sistema, aumentare la sicurezza e trovare opportunità di risparmio di denaro. Puoi anche disattivare le istanze non di produzione e utilizzare Amazon CloudWatch e Auto Scaling per far fronte agli aumenti o alle riduzioni della domanda.

Infine, puoi usare AWS Cost Explorer o Quick con il file AWS Cost and Usage Report (CUR) o i log delle applicazioni per eseguire un'analisi avanzata della domanda del carico di lavoro.

Nel complesso, un'analisi completa della domanda dei carichi di lavoro consente alle organizzazioni di prendere decisioni informate sul provisioning, il dimensionamento e l'ottimizzazione delle risorse, con conseguente miglioramento delle prestazioni, dell'efficienza dei costi e della soddisfazione degli utenti.

Passaggi dell'implementazione

- Analizza i carichi di lavoro esistenti: analizza i dati provenienti dal carico di lavoro esistente, dalle versioni precedenti del carico di lavoro o dai modelli di utilizzo previsti. Utilizza Amazon CloudWatch, i file di log e i dati di monitoraggio per ottenere informazioni dettagliate su come è stato utilizzato il carico di lavoro. Analizza un ciclo completo del carico di lavoro e raccogli i dati per eventuali variazioni stagionali, ad esempio eventi di fine mese o di fine anno. L'attività che emerge dall'analisi deve riflettere le caratteristiche del carico di lavoro. L'impegno maggiore dovrebbe riguardare i carichi di lavoro di alto valore che presentano le maggiori variazioni della domanda. Il minimo impegno dovrebbe riguardare carichi di lavoro di basso valore che hanno variazioni minime nella domanda.
- Prevedi le influenze esterne: incontra i membri del team di tutta l'organizzazione che possono influenzare o modificare la domanda del carico di lavoro. I team più comuni sono le vendite, il marketing o il business development. Collabora con loro per conoscere i cicli secondo cui operano e se ci sono eventi che potrebbero modificare la domanda del carico di lavoro. Prevedi la richiesta del carico di lavoro con questi dati.

Risorse

Documenti correlati:

- Amazon CloudWatch
- AWS Trusted Advisor
- AWS X-Ray
- AWS Auto Scaling
- AWS Instance Scheduler
- Getting started with Amazon SQS
- AWS Cost Explorer
- Rapidità

Esempi correlati:

- Monitor, Track and Analyze for cost optimization
- Searching and analyzing logs in CloudWatch

COST09-BP02 Implementazione di un buffer o della limitazione (della larghezza di banda della rete) per gestire la domanda

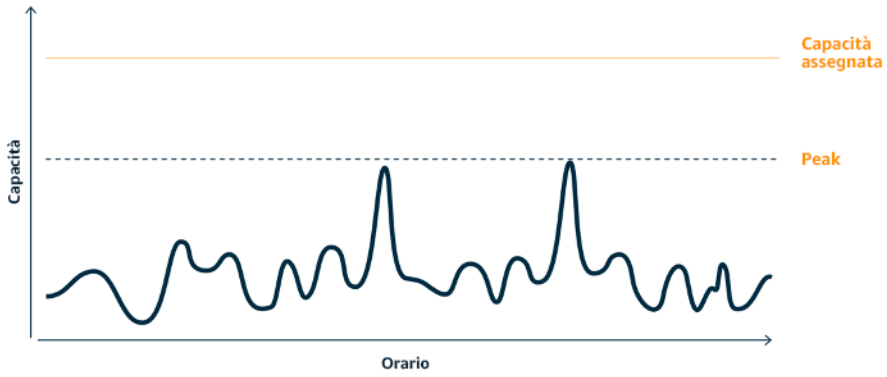
Il buffering e la limitazione (della larghezza di banda della rete) modificano la domanda sul carico di lavoro, attenuando eventuali picchi. Implementa la limitazione (della larghezza di banda della rete) quando i client eseguono nuovi tentativi. Implementa il buffering per archiviare la richiesta e rinviare l'elaborazione a un secondo momento. Verifica che le esecuzioni di limitazione (della larghezza di banda della rete) e buffering siano progettate in modo che i client ricevano una risposta nel tempo richiesto.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

L'implementazione di un buffer o di una limitazione (della larghezza di banda della rete) è fondamentale nel cloud computing per gestire la domanda e ridurre la capacità allocata richiesta per il carico di lavoro. Per ottenere prestazioni ottimali, è essenziale valutare la domanda totale, compresi i picchi, la velocità con cui variano le richieste e il tempo di risposta necessario. Quando i client hanno la possibilità di inviare nuovamente le proprie richieste, conviene applicare la limitazione (della larghezza di banda della rete). Al contrario, per i client che non dispongono della funzionalità di esecuzione di nuovi tentativi, l'approccio ideale è implementare una soluzione buffer. Tali buffer

semplificano l'afflusso di richieste e ottimizzano l'interazione delle applicazioni con diverse velocità operative.



Curva di domanda con due picchi distinti che richiedono un'elevata capacità allocata

Supponiamo che un carico di lavoro sia caratterizzato dalla curva della domanda illustrata nella figura precedente. Questo carico di lavoro presenta due picchi e per gestire tali picchi viene eseguito il provisioning della capacità di risorse mostrata dalla linea arancione. Le risorse e l'energia utilizzate per questo carico di lavoro non sono indicate nell'area sotto la curva della domanda, ma nell'area sotto la linea della capacità allocata, poiché per gestire questi due picchi è necessario eseguire il provisioning di tale capacità. Diminuire la curva della domanda del carico di lavoro può aiutarti a ridurre la capacità allocata di un carico di lavoro, oltre al suo impatto sull'ambiente. Per attenuare il picco, valuta la possibilità di implementare una soluzione basata sulla limitazione (della larghezza di banda della rete) o sul buffering.

Per comprendere meglio queste buffering e limitazione (della larghezza di banda della rete), proviamo ad analizzarle.

Limitazione (della larghezza di banda della rete): se l'origine della richiesta dispone di funzionalità di ripetizione dei tentativi, è possibile implementare la limitazione (della larghezza di banda della rete). La limitazione (della larghezza di banda della rete) indica all'origine che, se non è in grado di soddisfare la richiesta all'ora corrente, dovrebbe riprovare più tardi. L'origine attende un periodo di tempo, quindi riprova a eseguire la richiesta. L'implementazione della limitazione (della larghezza di banda della rete) ha il vantaggio di limitare la quantità massima di risorse e i costi del carico di lavoro. In AWS, puoi utilizzare Gateway Amazon API per implementare la limitazione (della larghezza di banda della rete).

Basato sul buffer: un approccio basato sul buffer si appoggia a produttori (componenti che inviano messaggi alla coda), consumatori (componenti che ricevono messaggi dalla coda) e una coda

(che contiene messaggi) per l'archiviazione dei messaggi. I messaggi vengono letti ed elaborati dai consumatori e ciò consente ai messaggi di essere eseguiti alla velocità che soddisfa i requisiti aziendali del consumatore stesso. Utilizzando una metodologia basata sul buffering, i messaggi dei produttori sono ospitati in code o flussi, dove i produttori possono accedervi a un ritmo in linea con le rispettive esigenze operative.

In AWS, puoi scegliere fra più servizi per l'implementazione di una strategia di buffering. Amazon Simple Queue Service (Amazon SQS) è un servizio gestito che offre code che consentono a un singolo consumatore di leggere singoli messaggi. Amazon Kinesis offre un flusso che consente a più consumatori di leggere gli stessi messaggi.

Il buffering e la limitazione (della larghezza di banda della rete) possono attenuare eventuali picchi modificando la domanda sul carico di lavoro. Usa la limitazione (della larghezza di banda della rete) quando i client riprovano le azioni e usa il buffering per bloccare la richiesta ed elaborarla in un secondo momento. Durante l'utilizzo dell'approccio basato sul buffering, assicurati di progettare il carico di lavoro per soddisfare la richiesta nel tempo richiesto e verifica di essere in grado di gestire le richieste duplicate. Analizza la domanda complessiva, la velocità di modifica e il tempo di risposta richiesto per determinare le dimensioni della limitazione (della larghezza di banda della rete) o del buffer richiesto.

Passaggi dell'implementazione

- Analizza i requisiti del client: analizza le richieste del client per determinare se sono in grado di eseguire nuovi tentativi. Per i client che non possono eseguire nuovi tentativi, è necessario implementare i buffer. Analizza la domanda complessiva, la velocità di modifica e il tempo di risposta richiesto per determinare le dimensioni della limitazione (della larghezza di banda della rete) o del buffer richiesto.
- Implementa un buffer o una limitazione (della larghezza di banda della rete): implementa un buffer o una limitazione (della larghezza di banda della rete) nel carico di lavoro. Una coda come Amazon Simple Queue Service (Amazon SQS) può fornire un buffer ai componenti del carico di lavoro. Gateway Amazon API è in grado di fornire la limitazione (della larghezza di banda della rete) per i componenti del carico di lavoro.

Risorse

Best practice correlate:

- SUS02-BP06 Implementazione del buffering o della limitazione (della larghezza di banda della rete) per ridurre la curva della domanda
- REL05-BP02 Richieste di limitazione (della larghezza di banda della rete)

Documenti correlati:

- AWS Auto Scaling
- AWS Instance Scheduler
- Gateway Amazon API
- Amazon Simple Queue Service
- Getting started with Amazon SQS
- Amazon Kinesis

Video correlati:

- Choosing the Right Messaging Service for Your Distributed App

Esempi correlati:

- Managing and monitoring API throttling in your workloads
- Throttling a tiered, multi-tenant REST API at scale using API Gateway
- Enabling Tiering and Throttling in a Multi-Tenant Amazon EKS SaaS Solution Using Amazon API Gateway
- Application integration Using Queues and Messages

COST09-BP03 Fornitura dinamica delle risorse

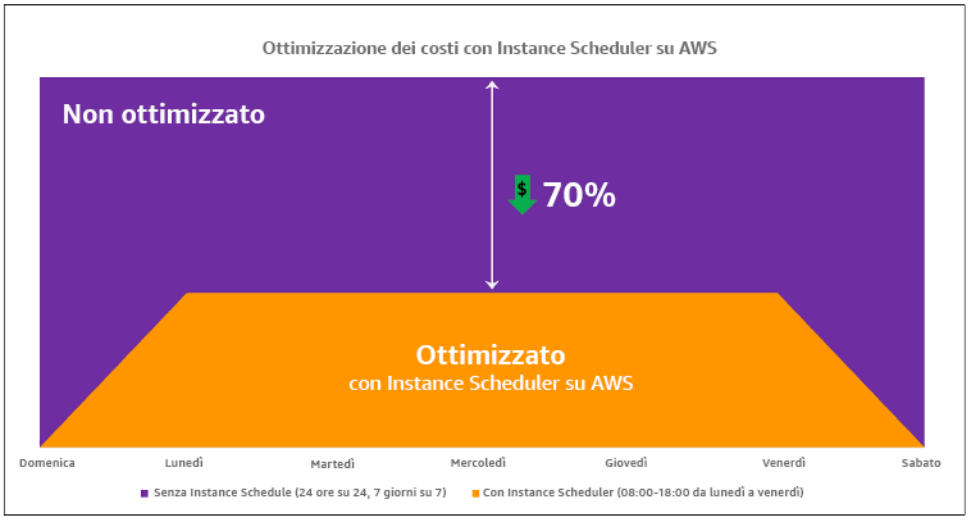
Le risorse sono allocate in modo pianificato. La pianificazione può essere basata sulla domanda, ad esempio tramite il dimensionamento automatico, oppure sul tempo, quando la domanda è prevedibile e le risorse sono fornite in base al tempo. Questi metodi comportano la minore quantità possibile di provisioning in eccesso o in difetto.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Esistono diversi modi in cui i clienti AWS possono aumentare le risorse disponibili per le proprie applicazioni e fornire risorse per soddisfare la domanda. Una di queste opzioni riguarda l'utilizzo di AWS Instance Scheduler per automatizzare l'avvio e l'interruzione delle istanze Amazon Elastic Compute Cloud (Amazon EC2) e Amazon Relational Database Service (Amazon RDS). L'altra opzione è utilizzare AWS Auto Scaling, che consente di scalare automaticamente le risorse di calcolo in base alla richiesta dell'applicazione o del servizio. Fornire risorse in base alla domanda ti consentirà di pagare solo per le risorse che usi, di ridurre i costi lanciando le risorse quando sono necessarie e di interromperle quando non servono più.

AWS Instance Scheduler consente di configurare l'arresto e l'avvio delle istanze Amazon EC2 e Amazon RDS a orari definiti, in modo da poter soddisfare la domanda delle stesse risorse secondo uno schema orario coerente, ad esempio ogni giorno gli utenti accedono alle istanze Amazon EC2 alle otto del mattino che non servono dopo le sei di sera. Questa soluzione aiuta a ridurre i costi operativi fermando le risorse non utilizzate e avviandole quando sono necessarie.



Ottimizzazione dei costi con AWS Instance Scheduler.

Puoi anche configurare in modo semplice e rapido le pianificazioni per le tue istanze Amazon EC2 nei tuoi account e nelle tue regioni con un'interfaccia utente (UI) utilizzando Configurazione rapida di AWS Systems Manager. Puoi pianificare le istanze Amazon EC2 o Amazon RDS con AWS Instance Scheduler e arrestare e avviare le istanze esistenti. Tuttavia, non puoi arrestare e avviare istanze presenti nel tuo gruppo Auto Scaling (ASG) o che gestiscono servizi come Amazon Redshift o il servizio OpenSearch di Amazon I gruppi Auto Scaling presentano una propria pianificazione in merito alle istanze del gruppo e queste istanze vengono create.

AWS Auto Scaling ti aiuta a regolare la capacità per mantenere prestazioni stabili e prevedibili al minor costo possibile per soddisfare le mutevoli esigenze. Si tratta di un servizio completamente gestito e gratuito per scalare la capacità della tua applicazione, integrato con istanze Amazon EC2 e parchi istanze spot, Amazon ECS, Amazon DynamoDB e Amazon Aurora. Auto Scaling fornisce il rilevamento automatico delle risorse per aiutare a trovare risorse nel carico di lavoro che possono essere configurate, dispone di strategie di dimensionamento integrate per ottimizzare le prestazioni, i costi o un equilibrio tra i due e fornisce il dimensionamento predittivo per aiutare a risolvere i picchi ricorrenti con regolarità.

Sono disponibili diverse opzioni di dimensionamento per scalare il tuo gruppo Auto Scaling:

- Mantenimento dei livelli di istanza correnti in qualsiasi momento
- Dimensionamento manuale
- Dimensionamento in base a una pianificazione
- Dimensionamento on demand
- Utilizzo del dimensionamento predittivo

Le policy di Auto Scaling sono diverse e possono essere classificate come policy di dimensionamento dinamico e pianificato. Le policy dinamiche fanno riferimento al dimensionamento manuale o dinamico, programmato o predittivo. È possibile utilizzare le policy di dimensionamento per il dimensionamento dinamico, pianificato e predittivo. Puoi inoltre utilizzare metriche e allarmi di Amazon CloudWatch per attivare eventi di dimensionamento per il tuo carico di lavoro. Noi ti suggeriamo di utilizzare i modelli di avvio, che consentono di accedere alle funzionalità e ai miglioramenti più recenti. In caso di utilizzo di configurazioni di avvio, non tutte le funzionalità di Auto Scaling sono disponibili. Ad esempio, non è possibile creare un gruppo Auto Scaling che avvii istanze spot e on demand oppure che specifichi più tipi di istanza. Per configurare queste caratteristiche, sarà necessario utilizzare un modello di avvio. Quando utilizzi i modelli di avvio, ti consigliamo di modificare ciascuno di essi. Con il controllo delle versioni dei modelli di avvio, è possibile creare un sottoinsieme del set completo di parametri. Quindi, è possibile riutilizzarlo per creare altre versioni dello stesso modello di avvio.

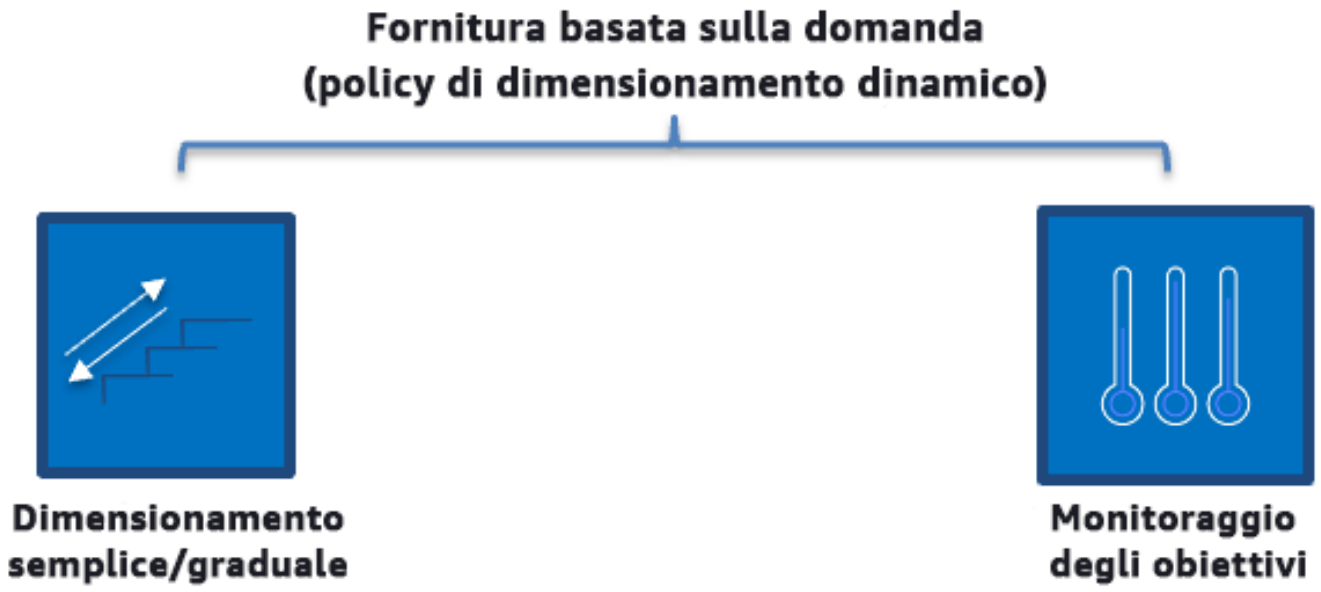
Puoi utilizzare AWS Auto Scaling o incorporare il ridurre orizzontalmente nel codice con API o SDK AWS. Ciò riduce i costi complessivi del carico di lavoro rimuovendo i costi operativi dall'apportare manualmente modifiche al tuo ambiente; le modifiche possono essere apportate molto più rapidamente. In questo modo, inoltre, il carico di lavoro viene adattato alla domanda in qualsiasi momento. Per seguire questa best practice e fornire risorse in modo dinamico all'organizzazione, è necessario comprendere la scalabilità verticale e orizzontale in Cloud AWS e la natura delle

applicazioni in esecuzione sulle istanze Amazon EC2. È meglio che il team di Cloud Financial Management collabori con i team tecnici per seguire questa best practice.

Elastic Load Balancing (bilanciamento del carico elastico) consente di scalare le risorse distribuendo la domanda su più risorse. Utilizzando ASG ed Elastic Load Balancing, puoi gestire le richieste in arrivo ottimizzando l'instradamento del traffico in modo che nessuna istanza venga sovraccaricata in un gruppo Auto Scaling. Le richieste vengono distribuite tra tutti gli obiettivi di un gruppo target in modalità Round Robin, senza tenere conto della capacità o dell'utilizzo.

Le metriche tipiche possono essere metriche standard di Amazon EC2, ad esempio l'utilizzo della CPU, il throughput, e la latenza di richiesta/risposta osservata da Elastic Load Balancing. Quando possibile, è consigliabile utilizzare un parametro indicativo dell'esperienza del cliente, in genere si tratta di un parametro personalizzato che potrebbe avere origine dal codice dell'applicazione all'interno del carico di lavoro. Per capire come soddisfare la domanda in modo dinamico in questo documento, Auto Scaling verrà suddiviso in due categorie (modello di fornitura basata sulla domanda e modello di fornitura basata sul tempo) e verrà approfondito ciascun modello.

Fornitura basata sulla domanda: sfrutta l'elasticità del cloud per fornire risorse in grado di soddisfare la domanda in continua evoluzione facendo riferimento allo stato della domanda quasi in tempo reale. Per la fornitura basata sulla domanda, utilizza API o funzionalità dei servizi per modificare in modo programmatico la quantità di risorse del cloud nella tua architettura. Ciò ti consente di scalare i componenti nella tua architettura e aumentare il numero di risorse durante i picchi di domanda per mantenere le prestazioni, nonché diminuire la capacità quando la domanda cala in modo da ridurre i costi.

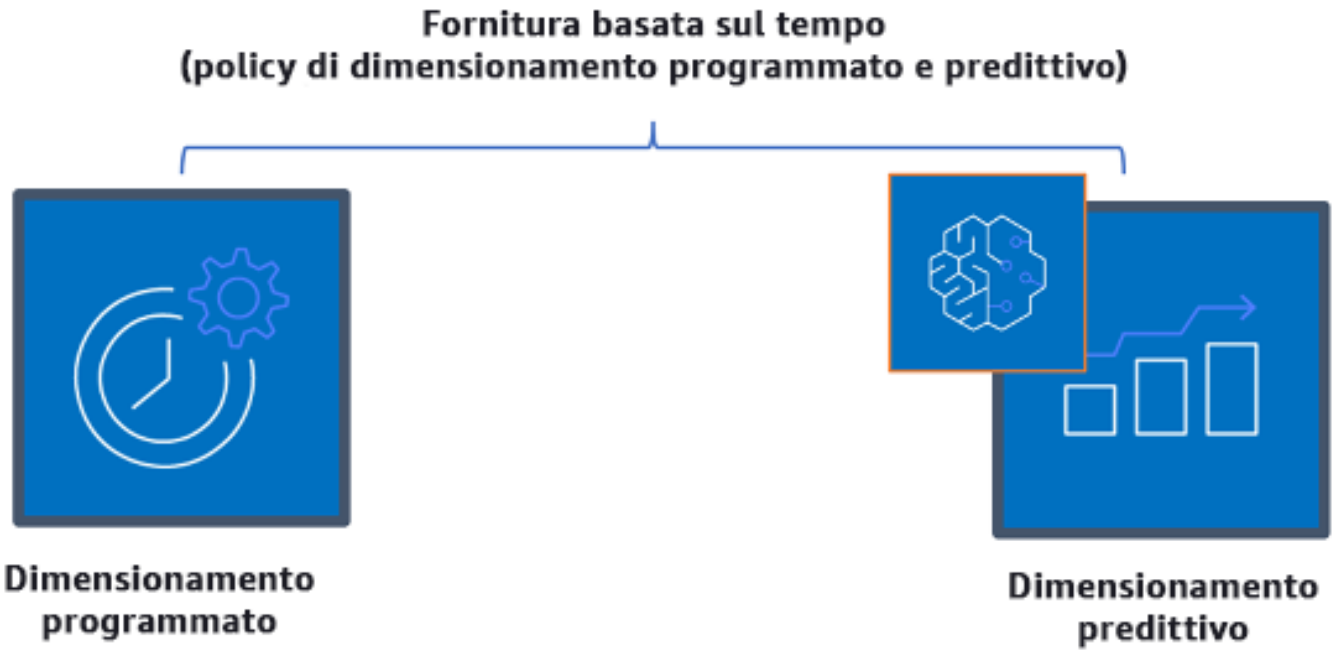


Policy di dimensionamento dinamico basato sulla domanda

- Dimensionamento semplice/graduale: monitora le metriche e aggiunge/rimuove le istanze secondo i passaggi definiti manualmente dai clienti.
- Monitoraggio degli obiettivi: meccanismo di controllo simile a un termostato che aggiunge o rimuove automaticamente le istanze per mantenere le metriche in base a un obiettivo definito dal cliente.

Quando prevedi una strategia basata sulla domanda in un progetto, tieni presenti due considerazioni principali. In primo luogo, devi capire con quale velocità è necessario predisporre le nuove risorse. In secondo luogo, devi capire che la dimensione del margine tra domanda e risorse fornite cambierà. Devi prepararti ad affrontare le variazioni nella domanda, nonché le risorse insufficienti.

Fornitura basata sul tempo: una strategia basata sul tempo allinea la capacità delle risorse alla domanda, che è prevedibile o ben definita nel tempo. In genere questa strategia non dipende dai livelli di utilizzo delle risorse. Una strategia basata sul tempo assicura che le risorse siano disponibili nel momento esatto in cui vengono richieste e possano essere fornite senza ritardi dovuti alle procedure di avvio e ai controlli di sistema o di coerenza. Attraverso una strategia basata sul tempo si possono fornire risorse aggiuntive o incrementare la capacità nei periodi più intensi.



Policy di dimensionamento basato sul tempo

Puoi utilizzare il dimensionamento automatico pianificato e predittivo per implementare un approccio basato sul tempo. I carichi di lavoro possono essere programmati per aumentare orizzontalmente in determinati momenti (ad esempio, all'inizio dell'orario di lavoro), garantendo quindi la disponibilità delle risorse all'arrivo degli utenti on demand. Il dimensionamento predittivo utilizza modelli per aumentare orizzontalmente, mentre il dimensionamento pianificato utilizza tempi predefiniti per aumentare orizzontalmente. Si può anche usare una strategia di selezione del tipo di istanza basata su attributi (ABS) nei gruppi Auto Scaling che consenta di esprimere i requisiti dell'istanza come un set di attributi, ad esempio vCPU, memoria e spazio di archiviazione. È possibile utilizzare automaticamente i tipi di istanza di nuova generazione quando vengono rilasciati e accedere a una gamma più ampia di capacità con le istanze spot di Amazon EC2. Amazon EC2 Fleet e Amazon EC2 Auto Scaling selezionano e avviano istanze che corrispondono agli attributi specificati, eliminando la necessità di scegliere manualmente i tipi di istanza.

Puoi anche sfruttare API e SDK AWS e AWS CloudFormation per allocare e disattivare automaticamente interi ambienti quando ne hai bisogno. Questa strategia risulta particolarmente adatta per gli ambienti di sviluppo o di prova che operano solo in determinati orari di lavoro o periodi di tempo. Puoi usare le API per scalare le risorse all'interno di un ambiente (scalabilità verticale). Ad esempio, potresti aumentare verticalmente un carico di lavoro di produzione modificando la dimensione o la classe dell'istanza. Ciò è possibile interrompendo e avviando l'istanza e selezionando una dimensione o classe diversa. Questa tecnica può essere applicata anche ad altre risorse, come i volumi elastici Amazon EBS, che possono essere modificati per aumentarne le dimensioni, regolarne le prestazioni (IOPS) o cambiare il tipo di volume durante l'utilizzo.

Quando prevedi una strategia basata sul tempo in un progetto, tieni presenti due considerazioni principali. In primo luogo, che livello di coerenza presenta il modello di utilizzo? In secondo luogo, qual è l'impatto se il modello cambia? Puoi migliorare l'accuratezza delle previsioni monitorando i tuoi carichi di lavoro e utilizzando la business intelligence. Se si notano cambiamenti significativi nel modello di utilizzo, si possono modificare i tempi per assicurarti che la copertura sia fornita.

Passaggi dell'implementazione

- Configura il dimensionamento pianificato: per le variazioni prevedibili della domanda, il dimensionamento basato sul tempo può fornire il numero corretto di risorse in modo tempestivo. È utile anche se la creazione e la configurazione delle risorse non avvengono in maniera sufficientemente rapida per rispondere alle modifiche on demand. Utilizzando l'analisi del carico di lavoro, configura il dimensionamento pianificato utilizzando AWS Auto Scaling. Per configurare

la pianificazione basata sul tempo, è possibile utilizzare il dimensionamento predittivo del dimensionamento pianificato per aumentare il numero di istanze Amazon EC2 nei gruppi Auto Scaling in anticipo in base alle variazioni di carico previste o prevedibili.

- Configura il dimensionamento predittivo: il dimensionamento predittivo ti consente di aumentare il numero di istanze Amazon EC2 nel gruppo Auto Scaling in anticipo rispetto ai modelli giornalieri e settimanali nei flussi di traffico. Se si hanno picchi di traffico regolari e applicazioni che richiedono molto tempo per avviarsi, si dovrebbe prendere in considerazione l'utilizzo del dimensionamento predittivo. Il dimensionamento predittivo può aiutare a scalare più velocemente inizializzando la capacità prima del carico previsto rispetto al solo dimensionamento dinamico, che è di natura reattiva. Ad esempio, se gli utenti iniziano a utilizzare il carico di lavoro all'inizio dell'orario di lavoro e non lo utilizzano dopo l'orario di lavoro, il dimensionamento predittivo può aggiungere capacità prima dell'orario di lavoro, eliminando i ritardi del dimensionamento dinamico per reagire alle variazioni del traffico.
- Configura il dimensionamento automatico dinamico: per configurare il dimensionamento in base ai parametri del carico di lavoro attivi, utilizza Auto Scaling. Utilizza l'analisi e configura Auto Scaling per l'avvio sui livelli di risorse corretti e assicurati che il carico di lavoro si riduca orizzontalmente nel tempo richiesto. È possibile avviare e scalare automaticamente un parco di istanze on demand e istanze spot all'interno di un singolo gruppo con un singolo gruppo Auto Scaling. Oltre a ricevere sconti per l'utilizzo di Istanze Spot, è possibile utilizzare Istanze riservate o Savings Plan per ricevere tariffe scontate sul normale prezzo delle istanze on demand. Tutti questi fattori insieme ti aiutano a risparmiare sui costi per le istanze Amazon EC2 e ti assicurano la scalabilità e le prestazioni desiderate per l'applicazione.

Risorse

Documenti correlati:

- AWS Auto Scaling
- AWS Instance Scheduler
- Dimensiona il gruppo Auto Scaling
- Getting Started with Amazon EC2 Auto Scaling
- Getting started with Amazon SQS
- Scheduled Scaling for Amazon EC2 Auto Scaling
- Predictive scaling for Amazon EC2 Auto Scaling

Video correlati:

- Target Tracking Scaling Policies for Auto Scaling
- AWS Instance Scheduler

Esempi correlati:

- Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet
- Optimizing Amazon Elastic Container Service for cost using scheduled scaling
- Predictive Scaling with Amazon EC2 Auto Scaling
- Come posso utilizzare Instance Scheduler con CloudFormation per pianificare le istanze Amazon EC2?

Ottimizzazione nel tempo

Questions

- COST 10. In che modo valuti i nuovi servizi?
- COST 11. Come valuti il costo dell'impegno?

COST 10. In che modo valuti i nuovi servizi?

Nel momento in cui AWS rilascia nuovi servizi e funzionalità, è buona prassi rivedere le decisioni esistenti relative all'architettura per verificare che continuino a essere le più convenienti.

Best practice

- COST10-BP01 Sviluppare un processo di revisione del carico di lavoro
- COST10-BP02 Rivedi e analizza regolarmente questo carico di lavoro

COST10-BP01 Sviluppare un processo di revisione del carico di lavoro

Sviluppa un processo che definisca i criteri e il processo per la revisione del carico di lavoro. L'impegno analitico deve riflettere il potenziale risultato. Ad esempio, i carichi di lavoro principali o i carichi di lavoro con un valore superiore al 10% della fattura sono analizzati trimestralmente oppure ogni sei mesi, mentre i carichi di lavoro inferiori al 10% sono analizzati annualmente.

Livello di rischio associato se questa best practice non fosse adottata: elevato

Guida all'implementazione

Per far sì che il carico di lavoro sia sempre efficiente in termini di costi, devi analizzarlo regolarmente per stabilire se ci sono opportunità di implementare nuovi servizi, funzionalità e componenti. Per garantire costi complessivi ridotti, il processo deve essere proporzionale al potenziale risparmio. Ad esempio, i carichi di lavoro che rappresentano il 50% della spesa complessiva devono essere esaminati con maggiore regolarità e più nel dettaglio rispetto ai carichi di lavoro che rappresentano il 5% della spesa complessiva. Prendi in considerazione qualsiasi fattore esterno o volatilità. Se il carico di lavoro serve una determinata area geografica o un segmento di mercato e viene previsto un cambiamento in tale area, revisioni più frequenti possono portare a risparmi sui costi. Un altro fattore in fase di revisione è rappresentato dall'impegno necessario per implementare le modifiche. Se i test e la convalida delle modifiche comportassero costi significativi, le revisioni dovrebbero essere meno frequenti.

Prendi in considerazione il costo nel lungo termine della manutenzione di componenti e risorse obsoleti e legacy, nonché dell'impossibilità di implementare in essi nuove funzionalità. L'attuale costo del test e della convalida potrebbe superare il vantaggio auspicato. Tuttavia, nel corso del tempo, il costo di apportare modifiche potrebbe crescere in modo significativo all'aumentare del divario tra il carico di lavoro e le tecnologie attuali, generando costi ancora maggiori. Ad esempio, il costo del passaggio a un nuovo linguaggio di programmazione potrebbe attualmente non risultare conveniente. Tuttavia, nel giro di cinque anni, il costo del personale qualificato per tale linguaggio potrebbe aumentare e, a causa dell'aumento del carico di lavoro, potresti dover trasferire un sistema ancora più grande al nuovo linguaggio, richiedendo sforzi ancora maggiori rispetto a prima.

Suddividi il carico di lavoro in componenti, assegna un costo ai componenti (una stima è sufficiente), quindi elenca i fattori (ad esempio, impegno richiesto e mercati esterni) accanto a ciascun componente. Utilizza questi indicatori per determinare una frequenza di revisione per ogni carico di lavoro. Ad esempio, potresti avere i server Web come un costo elevato, con un impegno di modifica ridotto e fattori esterni elevati, e da questo potrebbe derivare un'alta frequenza di revisione. Un database centrale può avere un costo medio, con un impegno di modifica elevato e un basso fattore esterno, e da questo potrebbe derivare una frequenza di revisione media.

Definisci un processo per valutare i nuovi servizi, i modelli di progettazione, i tipi di risorse e le configurazioni per ottimizzare il costo del tuo carico di lavoro man mano che diventano disponibili. Proprio come avviene nella revisione del pilastro delle prestazioni e nella revisione del pilastro dell'affidabilità, identifica, convalida e assegna la priorità alle attività di ottimizzazione e miglioramento e alla risoluzione dei problemi, quindi inseriscile nel tuo backlog.

Passaggi dell'implementazione

- Definisci la frequenza della revisione: definisci la frequenza con cui il carico di lavoro e i relativi componenti devono essere revisionati. Dedica tempo e risorse al miglioramento continuo e alla frequenza di revisione per migliorare l'efficienza e l'ottimizzazione del carico di lavoro. Si tratta di una combinazione di fattori e può variare da carico di lavoro a carico di lavoro all'interno dell'organizzazione, ma può anche variare tra i componenti del carico di lavoro. I fattori più comuni sono: l'importanza per l'organizzazione misurata in termini di fatturato o marchio, il costo totale di esecuzione del carico di lavoro (inclusi costi operativi e delle risorse), la complessità del carico di lavoro, la facilità di implementazione di una modifica, eventuali accordi di licenza software e l'eventuale aumento dei costi di licenza dovuti a licenze punitive in seguito a una modifica. I componenti possono essere definiti a livello funzionale o tecnico come server Web e database, oppure come risorse di calcolo e storage. Equilibra i fattori di conseguenza e prevedi un periodo per il carico di lavoro e i relativi componenti. Puoi decidere di rivedere l'intero carico di lavoro ogni 18 mesi, esaminare i server Web ogni 6 mesi, il database ogni 12 mesi, l'elaborazione e lo storage a breve termine ogni 6 mesi e lo storage a lungo termine ogni 12 mesi.
- Definisci la completezza della revisione: stabilisci quanto impegno deve essere impiegato per la revisione dei componenti o dell'intero carico di lavoro. Come per la frequenza di revisione, si tratta di un equilibrio tra più fattori. Valuta e dai priorità alle opportunità di miglioramento per concentrare gli sforzi dove producono i vantaggi maggiori, stimando l'impegno necessario per queste attività. Se i risultati non sono in linea con gli obiettivi e l'impegno richiesto ha un costo superiore, riprova utilizzando linee d'azione alternative. I processi di revisione devono prevedere l'allocazione di tempo e risorse per rendere possibile il miglioramento incrementale continuo. Ad esempio, si può decidere di dedicare una settimana all'analisi del componente del database, una settimana di analisi alle risorse di calcolo e quattro ore alla revisione dell'archiviazione.

Risorse

Documenti correlati:

- AWS Blog di notizie
- Tipi di cloud computing
- Novità di AWS

Esempi correlati:

- AWS Support Proactive Services
- Revisioni periodiche dei carichi di lavoro SAP

COST10-BP02 Rivedi e analizza regolarmente questo carico di lavoro

I carichi di lavoro esistenti vengono rivisti con regolarità in base a ogni processo definito per scoprire se è possibile adottare nuovi servizi, se i servizi esistenti possono essere sostituiti o se i carichi di lavoro possono essere riprogettati.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

AWS aggiunge costantemente nuove funzionalità in modo da poter sperimentare e innovare più velocemente con la tecnologia più recente. AWS What's New descrive in dettaglio come AWS si sta procedendo in tal senso e fornisce una rapida panoramica dei AWS servizi, delle funzionalità e degli annunci di espansione a livello regionale non appena vengono rilasciati. Puoi approfondire i rilasci previsti e usarli per la revisione e l'analisi dei tuoi carichi di lavoro esistenti. Per sfruttare i vantaggi dei nuovi AWS servizi e funzionalità, è necessario esaminare i carichi di lavoro e implementare nuovi servizi e funzionalità in base alle esigenze. Ciò significa che potrebbe essere necessario sostituire i servizi esistenti utilizzati per il carico di lavoro o modernizzare il carico di lavoro per adottare questi nuovi servizi. AWS Ad esempio, è possibile esaminare i carichi di lavoro e sostituire il componente di messaggistica con Amazon Simple Email Service. Ciò elimina il costo di gestione e manutenzione di un parco istanze, fornendo al contempo tutte le funzionalità a un costo ridotto.

Per analizzare il tuo carico di lavoro e individuare le opportunità potenziali, dovresti prendere in considerazione non solo i nuovi servizi, ma anche le nuove modalità per creare le soluzioni. Guarda i video di This is My Architecture su AWS per conoscere i progetti architettonici di altri clienti, le loro sfide e le loro soluzioni. Dai un'occhiata alla serie All-In per scoprire le applicazioni dei AWS servizi nel mondo reale e le storie dei clienti. Puoi inoltre guardare la serie di video Back to Basics che illustra, esamina e analizza le best practice di base relative ai modelli di architettura cloud. Un'altra fonte sono i video How to Build This, progettati per aiutare le persone con grandi idee su come dare vita al loro prodotto minimo redditizio (MVP) utilizzando AWS i servizi. È un modo per i costruttori di tutto il mondo che hanno una forte idea di ottenere indicazioni architettoniche da AWS Solutions Architects esperti. Infine, puoi consultare i materiali della risorsa Nozioni di base, che offre tutorial dettagliati.

Prima di avviare il processo di revisione segui i requisiti aziendali per il carico di lavoro, i requisiti sulla privacy dei dati e la sicurezza per usare un servizio o un'area geografica specifica e i requisiti di performance, seguendo al tempo stesso il processo di revisione concordato.

Passaggi dell'implementazione

- Rivedi con regolarità il carico di lavoro: utilizzando il processo definito, esegui le revisioni con la frequenza specificata. Accertati di dedicare la quantità di impegno necessaria per ciascun componente. Questo processo è simile a quello di progettazione iniziale in cui hai selezionato i servizi per l'ottimizzazione dei costi. Analizza i servizi e i vantaggi che porterebbero; questa volta considera anche il costo del tempo necessario per la modifica, non solo i vantaggi a lungo termine.
- Implementa nuovi servizi: se in seguito all'analisi ritieni di dover implementare modifiche, esegui innanzitutto una baseline del carico di lavoro per scoprire il costo corrente per ogni output. Implementa le modifiche, quindi esegui un'analisi per verificare il nuovo costo per ogni output.

Risorse

Documenti correlati:

- AWS Blog di notizie
- Novità di AWS
- AWS Documentazione
- AWS Guida introduttiva
- AWS Risorse generali

Video correlati:

- AWS - Questa è la mia architettura
- AWS - Ritorno alle basi
- AWS - Serie All-In
- How to Build This

COST 11. Come valuti il costo dell'impegno?

Best practice

- COST11-BP01 Esecuzione dell'automazione per le operazioni

COST11-BP01 Esecuzione dell'automazione per le operazioni

Valuta i costi operativi del cloud, concentrandoti sulla quantificazione del risparmio di tempo e impegno nelle attività amministrative e nelle implementazioni, sulla mitigazione del rischio di errore

umano, sulla conformità e su altre operazioni tramite l'automazione. Valuta il tempo e i costi associati necessari per gli impegni operativi e implementa l'automazione per le attività amministrative per ridurre al minimo il lavoro manuale laddove possibile.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

L'automazione delle operazioni riduce la frequenza di attività manuali, migliora l'efficienza e offre vantaggi ai clienti con un'esperienza affidabile e coerente durante l'implementazione, l'amministrazione o l'operatività dei carichi di lavoro. Puoi liberare le risorse dell'infrastruttura dalle attività operative manuali e usarle per operazioni e innovazioni di maggior valore, migliorando così i risultati aziendali. Le aziende vogliono un modo testato e collaudato di gestire i propri carichi di lavoro nel cloud. Tale soluzione deve essere sicura, veloce ed economica, con il minimo rischio e la massima affidabilità.

Inizia assegnando le priorità alle tue operazioni sulla base dell'impegno richiesto, considerando i costi complessivi. Ad esempio, quanto tempo è necessario per implementare nuove risorse nel cloud, eseguire modifiche di ottimizzazione alle risorse esistenti o implementare le configurazioni necessarie? Esamina il costo totale delle attività eseguite dal personale, tenendo conto dei costi operativi e di gestione. Dai la priorità alle automazioni per le attività amministrative per ridurre il livello di impegno delle persone.

L'impegno di revisione deve riflettere il potenziale risultato. Ad esempio, esamina il tempo impiegato per eseguire le attività manualmente rispetto a quello per eseguirle in automatico. Dai priorità all'automazione di attività ripetitive e di valore elevato che richiedono tempo e sono complesse. Le attività che presentano un rischio o un valore elevato di errore umano sono in genere il punto di partenza migliore da cui iniziare con l'automazione, poiché il rischio spesso comporta un costo operativo aggiuntivo indesiderato (come gli straordinari del team operativo).

Utilizza gli strumenti di automazione come AWS Systems Manager o AWS Config per semplificare le operazioni, la conformità, il monitoraggio, il ciclo di vita e i processi di terminazione. Con i servizi e gli strumenti AWS, nonché i prodotti di terze parti, puoi personalizzare le automazioni che implementi per soddisfare le tue esigenze specifiche. La tabella seguente mostra alcune delle funzioni e delle caratteristiche operative di base che puoi ottenere con i servizi AWS per automatizzare attività amministrative e operative:

- AWS Audit Manager: audit continuo dell'utilizzo di AWS per semplificare la valutazione di rischio e conformità.

- AWS Backup: gestione e automazione centralizzata della protezione dei dati.
- AWS Config: configurazione delle risorse di elaborazione, valutazione, audit, esame delle configurazioni e dell'inventario delle risorse.
- AWS CloudFormation: avvio di risorse ad alta disponibilità con il modello Infrastructure as Code.
- AWS CloudTrail: gestione, conformità e controllo delle modifiche IT.
- Amazon EventBridge: pianificazione di eventi e trigger AWS Lambda per le azioni.
- AWS Lambda: automatizzazione dei processi ripetitivi attivandoli con eventi o eseguendoli in base a una pianificazione fissa con AWS EventBridge.
- AWS Systems Manager: avvio e arresto dei carichi di lavoro, applicazione di patch ai sistemi operativi, automatizzazione di configurazione e gestione continua.
- AWS Step Functions: pianificazione di lavori e automazione dei flussi di lavoro.
- AWS Service Catalog: utilizzo dei modelli, modello Infrastructure as code con conformità e controllo.

Se desideri adottare da subito le automazioni utilizzando prodotti e servizi AWS e non disponi delle competenze giuste nella tua organizzazione, contatta AWS Managed Services (AMS), AWS Professional Services o i partner AWS per promuovere l'adozione dell'automazione e migliorare la tua eccellenza operativa nel cloud.

AWS Managed Services (AMS) è un servizio che gestisce l'infrastruttura AWS per conto di clienti e partner aziendali. Fornisce un ambiente sicuro e conforme in cui è possibile distribuire i carichi di lavoro. AMS utilizza modelli operativi cloud aziendali dotati di automazione per consentirti di soddisfare i requisiti aziendali, di passare più rapidamente al cloud e di ridurre i costi di gestione correnti.

AWS Professional Services può anche aiutarti a raggiungere i risultati aziendali auspicati e ad automatizzare le operazioni con AWS. Consente ai clienti di implementare operazioni IT automatizzate, solide e agili, nonché funzionalità di governance ottimizzate per il cloud. Per esempi di monitoraggio dettagliati e best practice consigliate, consulta il whitepaper sul pilastro dell'eccellenza operativa.

Passaggi dell'implementazione

- Sviluppa una volta e implementa più volte: utilizza il modello Infrastructure as code come CloudFormation, SDK AWS o AWS CLI per eseguire l'implementazione una sola volta e usarla più volte per ambienti simili o per casi di disaster recovery. Applica i tag durante l'implementazione per

monitorare il tuo consumo definito in altre best practice. Utilizza AWS Launch Wizard per ridurre i tempi di implementazione di molti carichi di lavoro aziendali più diffusi. AWS Launch Wizard ti guida nel ridimensionamento, nella configurazione e nell'implementazione dei carichi di lavoro aziendali seguendo le best practice AWS. Puoi anche usare Service Catalog per creare e gestire modelli Infrastructure as code approvati da utilizzare in AWS cosicché tutti possano scoprire risorse cloud self-service approvate.

- Automatizza la conformità continua: prendi in considerazione l'automazione di valutazioni e correzioni delle configurazioni registrate rispetto agli standard predefiniti. Grazie alla sinergia fra AWS Organizations e le funzionalità di AWS Config e AWS CloudFormation, puoi gestire e automatizzare in modo efficiente la conformità alla configurazione su larga scala per centinaia di account membri. Puoi esaminare le modifiche delle configurazioni e delle relazioni tra le risorse AWS e approfondire la cronologia di una configurazione di risorsa.
- Automatizza le attività di monitoraggio: AWS offre svariati strumenti di monitoraggio dei servizi. Puoi configurare questi strumenti per automatizzare le attività di monitoraggio. Crea e implementa un piano di monitoraggio che raccolga i dati da tutte le parti del carico di lavoro in modo da poter eseguire più facilmente il debug di un errore su più punti, se si verifica. Ad esempio, puoi utilizzare gli strumenti di monitoraggio automatizzati per osservare Amazon EC2 e ricevere una segnalazione quando qualcosa non va secondo i controlli dello stato del sistema, i controlli dello stato delle istanze e gli allarmi Amazon CloudWatch.
- Automatizza manutenzione e operazioni: esegui in automatico operazioni di routine senza l'intervento umano. Usando i servizi e gli strumenti AWS, puoi scegliere quali automazioni AWS implementare e personalizzare in base alle tue esigenze specifiche. Ad esempio, utilizza EC2 Image Builder per la creazione, il test e l'implementazione di immagini di macchine virtuali e container da utilizzare in AWS oppure on-premises o per l'applicazione di patch alle istanze EC2 con AWS SSM. Se l'azione desiderata non può essere eseguita con i servizi AWS o ti servono azioni più complesse con il filtraggio delle risorse, automatizza le operazioni con gli strumenti AWS Command Line Interface (AWS CLI) o AWS SDK. AWS CLI consente di automatizzare l'intero processo di controllo e gestione dei servizi AWS con script senza utilizzare Console di gestione AWS. Seleziona i tuoi AWS SDK preferiti per interagire con i servizi AWS. Per altri esempi di codice, consulta il repository di esempi di AWS SDK Code.
- Crea un ciclo di vita continuo con le automazioni: è importante stabilire e preservare policy consolidate del ciclo di vita non solo per le normative o la ridondanza, ma anche per l'ottimizzazione dei costi Puoi utilizzare AWS Backup per gestire e automatizzare centralmente la protezione dei dati degli archivi di dati, come bucket, volumi, database e file system. Puoi anche utilizzare Amazon Data Lifecycle Manager per automatizzare la creazione, la conservazione e l'eliminazione degli snapshot EBS e delle AMI EBS-backed.

- Elimina le risorse non necessarie: l'accumulo di risorse inutilizzate in sandbox o negli Account AWS di sviluppo avviene di frequente. Gli sviluppatori creano e sperimentano vari servizi e risorse come parte del normale ciclo di sviluppo, quindi non eliminano le risorse quando non sono più necessarie. Le risorse inutilizzate possono comportare costi superflui e talvolta elevati per l'organizzazione. L'eliminazione di queste risorse può ridurre i costi operativi di questi ambienti. Assicurati che i dati non siano necessari o esegui un backup se non sei sicuro. È possibile usare AWS CloudFormation per pulire gli stack implementati, eliminando automaticamente la maggior parte delle risorse definite nel modello. In alternativa, puoi creare un'automazione per l'eliminazione delle risorse AWS utilizzando strumenti come aws-nuke.

Risorse

Documenti correlati:

- Modernizing operations in the Cloud AWS
- AWS Services for Automation
- Infrastructure and automation
- AWS Systems Manager Automation
- Monitoraggio automatico e manuale
- AWS automations for SAP administration and operations
- AWS Managed Services
- AWS Professional Services

Video correlati:

- Automate Continuous Compliance at Scale in AWS
- AWS Backup Demo: Cross-Account & Cross-Region Backup
- Patching for your Amazon EC2 Instances

Esempi correlati:

- Reinventing automated operations (Part I)
- Reinventing automated operations (Part II)
- Automate deletion of AWS resources by using aws-nuke

- Delete unused Amazon EBS volumes by using AWS Config and AWS SSM
- Automate continuous compliance at scale in AWS
- IT Automations with AWS Lambda

Sostenibilità

Il pilastro della sostenibilità include la consapevolezza dell'impatto dei servizi utilizzati, la quantificazione di tale impatto per l'intero ciclo di vita del carico di lavoro e l'applicazione dei principi di progettazione e delle best practice per ridurlo nella fase di sviluppo di carichi di lavoro cloud. Puoi trovare linee guida prescrittive sull'implementazione nel whitepaper sul pilastro della sostenibilità.

Aree delle best practice

- Selezione della regione
- Allineamento alla domanda
- Software e architettura
- Dati
- Hardware e servizi
- Processo e cultura

Selezione della regione

Domanda

- SUS 1. Come si selezionano le regioni per un carico di lavoro?

SUS 1. Come si selezionano le regioni per un carico di lavoro?

La scelta della regione per il carico di lavoro influisce in modo significativo sui relativi KPI, tra cui prestazioni, costi e impatto ambientale. Per ottimizzare questi KPI, è necessario scegliere le regioni per i propri carichi di lavoro in base alle esigenze aziendali e agli obiettivi di sostenibilità.

Best practice

- SUS01-BP01 Scelta della regione in base alle esigenze aziendali e agli obiettivi di sostenibilità.

SUS01-BP01 Scelta della regione in base alle esigenze aziendali e agli obiettivi di sostenibilità.

Scegli la regione del tuo carico di lavoro in base alle esigenze aziendali e agli obiettivi di sostenibilità per ottimizzare i suoi KPI, tra cui prestazioni, costi e impatto ambientale.

Anti-pattern comuni:

- Selezione della regione del carico di lavoro in base alla propria collocazione.
- Consolidamento di tutte le risorse del carico di lavoro in un'unica posizione geografica.

Vantaggi dell'adozione di questa best practice: riduzione dell'impronta di carbonio di un carico di lavoro collocandolo vicino ai progetti legati alle energie rinnovabili di Amazon o alle regioni con un'intensità ridotta di emissione di anidride carbonica.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Il Cloud AWS è una rete in costante espansione di regioni e point of presence (POP), con un'infrastruttura di rete globale che li collega tra loro. La scelta della regione per il carico di lavoro influisce in modo significativo sui relativi KPI, tra cui prestazioni, costi e impatto ambientale. Per migliorare efficacemente questi KPI, è necessario scegliere le regioni per il proprio carico di lavoro in base alle esigenze aziendali e agli obiettivi di sostenibilità.

Passaggi dell'implementazione

- Selezione delle potenziali Regioni - Segui questi passaggi per valutare e selezionare le potenziali Regioni per il tuo carico di lavoro in base ai requisiti aziendali, tra cui la conformità, le funzionalità disponibili, il costo e la latenza:
  - Verifica che queste Regioni siano conformi in base alle normative locali richieste (ad esempio, quelle sulla sovranità dei dati).
  - Consulta gli elenchi dei servizi AWS per regione per verificare la presenza nelle regioni di servizi e funzionalità adeguati alla gestione del tuo carico di lavoro.
  - Calcola il costo del carico di lavoro per ciascuna regione mediante il Calcolatore dei prezzi AWS.
  - Valuta la latenza di rete tra le sedi degli utenti finali e ogni Regione AWS.
- Scelta delle Regioni: scegli le Regioni in prossimità dei progetti di generazione di energia rinnovabile di Amazon e le Regioni in cui la griglia presenta un'intensità di emissione di anidride carbonica nota inferiore a quella di altre sedi (o Regioni).

- Individua linee guida sulla sostenibilità pertinenti per monitorare e confrontare le emissioni di carbonio su base annua in conformità al Greenhouse Gas Protocol (metodi basati su mercato e posizione).
- Scegli la regione in base al metodo utilizzato per monitorare le emissioni di anidride carbonica. Per ulteriori informazioni circa la scelta di una regione in base alle tue linee guida sulla sostenibilità, consulta How to select a Region for your workload based on sustainability goals.

Risorse

Documenti correlati:

- Understanding your carbon emission estimations
- Amazon Around the Globe
- Renewable Energy Methodology
- What to Consider when Selecting a Region for your Workloads

Video correlati:

- AWS re:Invent 2023 - Sustainability innovation in AWS Global Infrastructure
- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2022 - Architecting sustainably and reducing your AWS carbon footprint
- AWS re:Invent 2022 - Sustainability in AWS global infrastructure

Allineamento alla domanda

Domanda

- SUS 2. Come si allineano le risorse cloud alla domanda?

SUS 2. Come si allineano le risorse cloud alla domanda?

Il modo in cui gli utenti e le applicazioni utilizzano i tuoi carichi di lavoro e altre risorse può aiutarti a identificare i miglioramenti da implementare per raggiungere gli obiettivi di sostenibilità. Puoi scalare l'infrastruttura in modo che sia costantemente adatta alla domanda e verifica di usare solo le risorse minime necessarie per supportare gli utenti. Allinea i livelli di servizio alle esigenze dei clienti. Colloca

le risorse in modo da limitare la rete necessaria per il loro consumo da parte di utenti e applicazioni. Rimuovi gli asset inutilizzati. Offri ai membri del team dispositivi in grado di soddisfarne le esigenze con un impatto minimo in termini di sostenibilità.

Best practice

- SUS02-BP01 Scalare dinamicamente l'infrastruttura dei carichi di lavoro
- SUS02-BP02 Allinearsi agli obiettivi di sostenibilità SLAs
- SUS02-BP03 Interruzione della creazione e della manutenzione di risorse inutilizzate
- SUS02-BP04 Ottimizza il posizionamento geografico dei carichi di lavoro in base ai requisiti di rete
- SUS02-BP05 Ottimizzazione delle risorse dei membri del team in base alle attività eseguite
- SUS02-BP06 Implementare il buffering o il throttling per appiattire la curva di domanda

SUS02-BP01 Scalare dinamicamente l'infrastruttura dei carichi di lavoro

Usa l'elasticità del cloud e dimensiona la tua infrastruttura in modo dinamico per rispondere alla richiesta di fornitura di risorse cloud ed evitare il provisioning eccessivo nel tuo carico di lavoro.

Anti-pattern comuni:

- Mancato dimensionamento dell'infrastruttura in base al carico degli utenti.
- Costante dimensionamento manuale dell'infrastruttura.
- Dopo un evento di dimensionamento, lasci una capacità aumentata anziché ridurre il dimensionamento.

Vantaggi dell'adozione di questa best practice: configurazione e test dell'elasticità del carico di lavoro consentono di abbinare in modo ottimale l'offerta di risorse cloud alla domanda ed evitare capacità con un provisioning eccessivo. Puoi sfruttare i vantaggi dell'elasticità nel cloud per scalare automaticamente la capacità durante e dopo i picchi di richiesta ed essere sicuro di utilizzare solo il numero esatto di risorse necessario per soddisfare le esigenze aziendali.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Il cloud offre la flessibilità necessaria per espandere o ridurre le risorse in modo dinamico attraverso una serie di meccanismi per soddisfare i cambiamenti della domanda. La corrispondenza ottimale tra offerta e domanda consente l'impatto ambientale più basso per un carico di lavoro.

La domanda può essere fissa o variabile e richiede parametri e automazione, allo scopo di garantire che la gestione non diventi particolarmente onerosa. Le applicazioni possono essere scalate verticalmente (verso l'alto o verso il basso) modificando la dimensione dell'istanza, orizzontalmente (aumentando o diminuendo) modificando il numero di istanze o tramite una combinazione delle due opzioni.

Puoi adottare varie strategie di approccio per associare l'offerta di risorse alla domanda.

- Approccio al tracciamento degli obiettivi: monitora il parametro di dimensionamento e aumenta o diminuisci automaticamente la capacità in base alle esigenze.
- Dimensionamento predittivo: procedi a ridurre orizzontalmente in previsione delle tendenze giornaliere e settimanali.
- Approccio basato sulla pianificazione: imposta il tuo programma di dimensionamento in base alle variazioni di carico prevedibili.
- Scalabilità dei servizi: scegli servizi (come il serverless) dotati dimensionamento nativo per progettazione o con dimensionamento automatico come funzionalità.

Identifica i periodi di utilizzo assente o ridotto e dimensiona le risorse per evitare capacità in eccesso e migliorare il livello di efficienza.

Passaggi dell'implementazione

- L'elasticità corrisponde all'offerta di risorse disponibili rispetto alla relativa domanda. Istanze, container e funzioni offrono meccanismi di elasticità, sia insieme al dimensionamento automatico sia come funzionalità del servizio. AWS offre una gamma di meccanismi di dimensionamento automatico per avere la certezza che sia possibile procedere a ridurre verticalmente i carichi di lavoro in modo facile e veloce nei periodi di basso carico di utenti. Ecco alcuni esempi di meccanismi di dimensionamento automatico:

Meccanismo di dimensionamento automatico	Dove usarlo
Amazon EC2 Auto Scaling	Da utilizzare per verificare che sia disponibile il numero corretto di istanze Amazon EC2 per gestire il carico degli utenti dell'applicazione.
Application Auto Scaling	Da utilizzare per scalare in automatico le risorse per singoli servizi AWS oltre Amazon

Meccanismo di dimensionamento automatico	Dove usarlo
	EC2, ad esempio, funzioni Lambda o servizi Amazon Elastic Container Service (Amazon ECS).
Kubernetes Cluster Autoscaler	Da utilizzare per scalare automaticamente i cluster Kubernetes su AWS.

- Si parla spesso di dimensionamento con servizi di calcolo come le istanze Amazon EC2 o le funzioni AWS Lambda. Prendi in considerazione la configurazione di servizi non di calcolo, come le unità di capacità di lettura e scrittura di Amazon DynamoDB o le partizioni del flusso di dati Amazon Kinesis per soddisfare la domanda.
- Verifica che le metriche per l'aumento verticale o orizzontale siano convalidate in base al tipo di carico di lavoro implementato. Se implementi un'applicazione di transcodifica video, è previsto il 100% di utilizzo della CPU e non deve essere il parametro principale. Se necessario, puoi servirti di una metrica personalizzata (ad esempio, l'utilizzo della memoria) per la policy di dimensionamento. Per scegliere la metrica corretta, consulta le linee guida seguenti per Amazon EC2:
  - La metrica deve essere una metrica di utilizzo valida e descrivere il livello di impiego di un'istanza.
  - Il valore del parametro deve aumentare e diminuire in proporzione al numero di istanze nel gruppo con scalabilità automatica.
- Usa il dimensionamento dinamico anziché il dimensionamento manuale per il tuo gruppo Auto Scaling. È consigliabile utilizzare le policy di dimensionamento del monitoraggio degli obiettivi nel dimensionamento dinamico
- Verifica che le implementazioni dei carichi di lavoro siano in grado di aumentare orizzontalmente e ridurre orizzontalmente. Crea scenari di test per eventi in cui si procede a ridurre orizzontalmente per verificare che il carico di lavoro si comporti secondo le aspettative e che non incida sull'esperienza utente (come nel caso della perdita di sessioni persistenti). Ad esempio, puoi usare la cronologia delle attività per verificare le attività di dimensionamento per un gruppo Auto Scaling.
- Analizza il tuo carico di lavoro per individuare modelli prevedibili e dimensionare le tue risorse in modo proattivo, anticipando variazioni nella domanda previste e pianificate. Con il dimensionamento predittivo puoi eliminare la necessità di offrire capacità in eccedenza. Per ulteriori informazioni, consulta Dimensionamento predittivo con Amazon EC2 Auto Scaling.

Risorse

Documenti correlati:

- Getting Started with Amazon EC2 Auto Scaling
- Predictive Scaling for EC2, Powered by Machine Learning
- Analyze user behavior using Amazon OpenSearch Service, Amazon Data Firehose and Kibana
- What is Amazon CloudWatch?
- Monitoring DB load with Performance Insights on Amazon RDS
- Introducing Native Support for Predictive Scaling with Amazon EC2 Auto Scaling
- Introducing Karpenter - An Open-Source, High-Performance Kubernetes Cluster Autoscaler
- Deep Dive on Amazon ECS Cluster Auto Scaling

Video correlati:

- AWS re:Invent 2023 - Scaling on AWS for the first 10 million users
- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Build a cost-, energy-, and resource-efficient compute environment
- AWS re:Invent 2022 - Scaling containers from one user to millions
- AWS re:Invent 2023 - Scaling FM inference to hundreds of models with Amazon SageMaker AI
- AWS re:Invent 2023 - Harness the power of Karpenter to scale, optimize & upgrade Kubernetes

Esempi correlati:

- Autoscaling

SUS02-BP02 Allinearsi agli obiettivi di sostenibilità SLAs

Rivedi e ottimizza gli accordi sui livelli di servizio del carico di lavoro (SLA) in base agli obiettivi di sostenibilità per ridurre al minimo le risorse necessarie per supportare il carico di lavoro continuando a soddisfare le esigenze aziendali.

Anti-pattern comuni:

- I carichi di lavoro SLAs sono sconosciuti o ambigui.

- Sei tu a definire i tuoi SLA obiettivi in termini di disponibilità e prestazioni.
- Usi lo stello modello di progettazione (come l'architettura multi-AZ) per tutti i carichi di lavoro.

Vantaggi derivanti dall'adozione di questa best practice: l'allineamento SLAs agli obiettivi di sostenibilità porta a un utilizzo ottimale delle risorse soddisfacendo al contempo le esigenze aziendali.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

SLAsdefinisci il livello di servizio previsto da un carico di lavoro cloud, ad esempio tempi di risposta, disponibilità e conservazione dei dati. Questi influenzano l'architettura, l'utilizzo delle risorse e l'impatto ambientale di un carico di lavoro nel cloud. A cadenza regolare, rivedi SLAs e fai dei compromessi che riducano in modo significativo l'utilizzo delle risorse in cambio di riduzioni accettabili dei livelli di servizio.

Passaggi dell'implementazione

- Analizza gli obiettivi di sostenibilità: individua gli obiettivi di sostenibilità della tua organizzazione, come la riduzione delle emissioni di carbonio o l'ottimizzazione dell'utilizzo delle risorse.
- RevisioneSLAs: valuta le tue SLAs per valutare se soddisfano i tuoi requisiti aziendali. Se stai superando i limitiSLAs, esegui un'ulteriore revisione.
- Analizza i compromessi: esamina i compromessi in termini di complessità del carico di lavoro (come un elevato volume di utenti simultanei), prestazioni (come la latenza) e impatto sulla sostenibilità (come le risorse richieste). In genere, dare la priorità a due fattori va a scapito del terzo.
- Adeguamento SLAs: aggiusta la SLAs situazione adottando compromessi che riducano in modo significativo gli impatti sulla sostenibilità in cambio di riduzioni accettabili dei livelli di servizio.
  - Sostenibilità e affidabilità: i carichi di lavoro a elevata disponibilità presentano la tendenza a un maggiore consumo di risorse.
  - Sostenibilità e prestazioni: l'utilizzo di più risorse per aumentare le prestazioni potrebbe tradursi in un maggiore impatto ambientale.
  - Sostenibilità e sicurezza: carichi di lavoro eccessivamente sicuri potrebbero avere un impatto ambientale maggiore.
- Definisci la sostenibilità, SLAs se possibile: includi la sostenibilità nel tuo carico di SLAs lavoro. Ad esempio, definisci un livello minimo di utilizzo come sostenibilità SLA per le tue istanze di calcolo.

- Utilizza modelli di progettazione efficienti: utilizza modelli di progettazione come i microservizi per dare priorità alle AWS funzioni aziendali critiche e consentire livelli di servizio inferiori (come obiettivi in termini di tempi di risposta o tempi di ripristino) per funzioni non critiche.
- Comunica e stabilisci la responsabilità: condividi le informazioni SLAs con tutte le parti interessate, inclusi il team di sviluppo e i clienti. Utilizza i report per tracciare e monitorare ilSLAs. Assegna la responsabilità per raggiungere i tuoi obiettivi di sostenibilità. SLAs
- Utilizza incentivi e premi: utilizza incentivi e premi per raggiungere o superare SLAs gli obiettivi di sostenibilità in linea con gli obiettivi di sostenibilità.
- Revisione e iterazione: rivedi e modifica regolarmente i tuoi obiettivi SLAs per assicurarti che siano in linea con l'evoluzione degli obiettivi di sostenibilità e prestazioni.

Risorse

Documenti correlati:

- Understand resiliency patterns and trade-offs to architect efficiently in the cloud
- Importance of Service Level Agreement for SaaS Providers

Video correlati:

- AWS re:Invent 2023 - Capacità, disponibilità, efficienza dei costi: scegline tre
- AWS re:Invent 2023 - Architettura sostenibile: passato, presente e futuro
- AWS re:Invent 2023 - Modelli di integrazione avanzati e compromessi per sistemi liberamente accoppiati
- AWS re:Invent 2022 - Fornire architetture sostenibili e ad alte prestazioni
- AWS re:Invent 2022 - Crea un ambiente di elaborazione efficiente in termini di costi, energia e risorse

SUS02-BP03 Interruzione della creazione e della manutenzione di risorse inutilizzate

Disattiva le risorse non utilizzate nel tuo carico di lavoro per ridurre il numero di risorse cloud richieste per supportare la domanda e per ridurre gli sprechi.

Anti-pattern comuni:

- Non analizzi la tua applicazione per individuare le risorse ridondanti o non più necessarie.

- Non rimuovi le risorse ridondanti o non più necessarie.

Vantaggi dell'adozione di questa best practice: la rimozione delle risorse non utilizzati libera risorse e migliora l'efficienza complessiva del carico di lavoro cloud.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all’implementazione

Le risorse inutilizzate consumano risorse cloud come spazio di archiviazione e potenza di elaborazione. Individuando ed eliminando queste risorse, puoi liberare capacità e ottenere un'architettura cloud più efficiente. Analizza le risorse delle applicazioni con regolarità (come report precompilati, set di dati, immagini statiche e modelli di accesso alle risorse) per identificare ridondanze, sottoutilizzi e obiettivi potenziali di disattivazione. Elimina le risorse ridondanti per ridurre gli sprechi nel tuo carico di lavoro.

Passaggi dell'implementazione

- Predisponi un inventario: redigi un inventario completo al fine di individuare tutte le risorse all'interno del tuo carico di lavoro.
- Analizza l'utilizzo: usa strumenti di monitoraggio per identificare risorse statiche non più necessarie.
- Rimuovi le risorse inutilizzate: predisponi un piano per la rimozione delle risorse non più necessarie.
  - Prima di rimuovere qualsiasi risorsa, valuta l'impatto della rimozione sull'architettura.
  - Analizza le risorse generate in sovrapposizione per rimuovere le elaborazioni ridondanti.
  - Aggiorna le tue applicazioni per smettere di produrre e archiviare risorse che non sono più necessarie.
- Comunica con le terze parti: indica alle terze parti di smettere di produrre e di archiviare per tuo conto risorse gestite non più necessarie. Chiedi di consolidare le risorse ridondanti.
- Usa le policy del ciclo di vita: serviti delle policy del ciclo di vita per l'eliminazione in automatico le risorse inutilizzate.
  - Puoi utilizzare il ciclo di vita Amazon S3 per gestire gli oggetti durante il loro ciclo di vita.
  - È possibile utilizzare Amazon Data Lifecycle Manager per automatizzare la creazione, la conservazione e l'eliminazione degli snapshot Amazon EBS e delle AMI supportate da Amazon EBS.

- Rivedi e ottimizza: esamina con regolarità il tuo carico di lavoro per individuare e rimuovere risorse non utilizzate.

Risorse

Documenti correlati:

- Optimizing your AWS Infrastructure for Sustainability, Part II: Storage
- How do I terminate active resources that I no longer need on my Account AWS?

Video correlati:

- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Preserving and maximizing the value of digital media assets using Amazon S3
- AWS re:Invent 2023 - Optimize costs in your multi-account environments

SUS02-BP04 Ottimizza il posizionamento geografico dei carichi di lavoro in base ai requisiti di rete

Seleziona le sedi cloud e i servizi per il carico di lavoro per ridurre la distanza che il traffico di rete deve percorrere e diminuire così le risorse totali di rete richieste per supportare il carico di lavoro.

Anti-pattern comuni:

- Selezione della regione del carico di lavoro in base alla propria collocazione.
- Consolidamento di tutte le risorse del carico di lavoro in un'unica posizione geografica.
- Tutto il traffico passa attraverso i data center esistenti.

Vantaggi dell'adozione di questa best practice: il posizionamento di un carico di lavoro in prossimità dei relativi utenti garantisce la latenza più bassa possibile e la contemporanea riduzione del trasferimento dei dati nella rete e dell'impatto ambientale.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

L' Cloud AWS infrastruttura è costruita attorno a opzioni di localizzazione come Regioni, Zone di disponibilità, gruppi di collocamento e edge location come AWS OutpostsAWSLocal Zones. Queste

opzioni relative alle sedi sono responsabili della gestione della connettività tra i componenti delle applicazioni, i servizi cloud, le reti edge e i data center on-premises.

Analizza i modelli di accesso alla rete nel tuo carico di lavoro per stabilire come usare queste opzioni relative alle sedi cloud e ridurre la distanza che il traffico di rete deve percorrere.

Passaggi dell'implementazione

- Analizza i modelli di accesso alla rete nel tuo carico di lavoro per capire come gli utenti usano la tua applicazione.
  - Utilizza strumenti di monitoraggio, come Amazon CloudWatch e AWS CloudTrail, per raccogliere dati sulle attività di rete.
  - Analizza i dati per identificare il modello di accesso alla rete.
- Seleziona le regioni appropriate per l'implementazione del carico di lavoro in base ai seguenti elementi chiave:
  - Il tuo obiettivo di sostenibilità: come illustrato nella sezione Selezione della regione.
  - Ubicazione dei dati per le applicazioni a uso intensivo di dati, ad esempio applicazioni di big data e machine learning, il codice dell'applicazione dovrebbe essere eseguito il più vicino possibile ai dati.
  - Ubicazione degli utenti: per le applicazioni rivolte agli utenti, scegli una regione o più regioni vicine agli utenti del carico di lavoro.
  - Altri vincoli: prendi in considerazione vincoli, come costi e conformità, come illustrato in What to Consider when Selecting a Region for your Workloads.
- Usa la cache locale o le soluzioni di caching AWS per i dati di frequente utilizzo per migliorare le performance, ridurre lo spostamento dei dati e minimizzare l'impatto ambientale.

Servizio	Quando usare
Amazon CloudFront	Utilizzalo per memorizzare nella cache contenuti statici come immagini, script e video, nonché contenuti dinamici come API risposte o applicazioni web.
Amazon ElastiCache	Usalo per memorizzare nella cache i contenuti per le applicazioni Web.

Servizio	Quando usare
DynamoDB Accelerator	Usalo per aggiungere accelerazione in memoria alle tabelle DynamoDB.

- Utilizza servizi in grado di supportarti nell'esecuzione del codice in posizioni più vicine agli utenti del carico di lavoro:

Servizio	Quando usare
Lambda@Edge	Usalo per operazioni a uso intensivo di risorse di calcolo eseguite quando gli oggetti non si trovano nella cache.
CloudFront Funzioni Amazon	Utilizzalo per casi d'uso semplici come HTTP manipolazioni di richieste o risposte che possono essere avviate da funzioni di breve durata.
AWS IoT Greengrass	Usale per eseguire la memorizzazione nella cache di risorse di calcolo, messaggistica e dati per i dispositivi connessi.

- Utilizza il pooling delle connessioni per consentire il loro riutilizzo e ridurre le risorse richieste.
- Utilizza archivi di dati distribuiti che non si affidano a connessioni persistenti e aggiornamenti sincroni per garantire coerenza e servire le popolazioni regionali.
- Sostituisci la capacità di rete statica preallocata con una capacità dinamica condivisa e condividi l'impatto in termini di sostenibilità della capacità di rete con altri abbonati.

Risorse

Documenti correlati:

- Ottimizzazione dell' AWS infrastruttura per la sostenibilità, parte: rete III
- ElastiCache Documentazione Amazon
- Che cos'è Amazon CloudFront?
- Caratteristiche CloudFront principali di Amazon

- AWS Infrastruttura globale
- AWS Local Zones e AWS Outposts scelta della tecnologia giusta per il tuo carico di lavoro edge
- Placement groups
- AWS Local Zones
- AWS Outposts

Video correlati:

- Demistificazione del trasferimento di dati su AWS
- Scalabilità delle prestazioni di rete sulle istanze Amazon di nuova generazione EC2
- AWS Video esplicativo su Local Zones
- AWS Outposts: Overview and How it Works
- AWS re:Invent 2023 - Una strategia di migrazione per carichi di lavoro edge e locali
- AWS re:Invent 2021 -: Portare l'esperienza in sede AWS OutpostsAWS
- AWS re:Invent 2020 - AWS Wavelength: Esegui app con latenza ultra bassa sull'edge 5G
- AWS re:Invent 2022 - AWS Local Zones: creazione di applicazioni per un edge distribuito
- AWS re:Invent 2021 - Creazione di siti Web a bassa latenza con Amazon CloudFront
- AWS re:Invent 2022 - Migliora le prestazioni e la disponibilità con AWS Global Accelerator
- AWS re:Invent 2022 - Costruisci la tua rete WAN utilizzando AWS
- AWS re:Invent 2020: gestione globale del traffico con Amazon Route 53

Esempi correlati:

- AWS Workshop di networking
- Architecting for sustainability - Minimize data movement across networks

SUS02-BP05 Ottimizzazione delle risorse dei membri del team in base alle attività eseguite

Ottimizza le risorse fornite ai membri del team per ridurre al minimo l'impatto sulla sostenibilità ambientale e supportare al tempo stesso le loro esigenze.

Anti-pattern comuni:

- Ignori l'impatto dei dispositivi utilizzati dai membri del tuo team sull'efficienza complessiva della tua applicazione cloud.
- Gestisci e aggiorni manualmente le risorse utilizzate dai membri del tuo team.

Vantaggi dell'adozione di questa best practice: migliore efficienza complessiva delle applicazioni abilitate per il cloud grazie all'ottimizzazione delle risorse dei membri del team.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Identifica le risorse che i membri del tuo team usano per accedere ai tuoi servizi, il loro ciclo di vita atteso e l'impatto finanziario e di sostenibilità. Implementa strategie per ottimizzare queste risorse. Esegui ad esempio operazioni complesse, come rendering e compilazione, su infrastrutture scalabili altamente utilizzate, invece che su sistemi per utenti singoli, sottoutilizzati e con un alto dispendio energetico.

Passaggi dell'implementazione

- Utilizza workstation efficienti dal punto di vista energetico: fornisci ai membri del team workstation e periferiche efficienti dal punto di vista energetico. Utilizza in questi dispositivi funzionalità di gestione dell'alimentazione efficienti, come la modalità di risparmio energetico, per ridurne il consumo di energia.
- Usa la virtualizzazione: usa desktop virtuali e lo streaming di applicazioni per limitare gli aggiornamenti e i requisiti dei dispositivi.
- Favorisci la collaborazione remota: incoraggia i membri del team a servirsi di strumenti di collaborazione remota come Amazon Chime o AWS Wickr al fine di ridurre la necessità di spostamenti e le emissioni di carbonio associate.
- Usa software a basso consumo energetico: fornisci ai membri del team software a basso consumo energetico, procedendo a rimuovere o disattivare funzionalità e processi non necessari.
- Gestisci i cicli di vita: valuta l'impatto di processi e sistemi sul ciclo di vita dei tuoi dispositivi e seleziona soluzioni che riducono al minimo i requisiti per la sostituzione dei dispositivi, pur continuando a soddisfare i requisiti di business. Effettua regolarmente la manutenzione e l'aggiornamento delle workstation o del software per conservare e migliorare l'efficienza.
- Gestione remota dei dispositivi: implementa la gestione remota dei dispositivi per ridurre gli spostamenti aziendali.

- AWS Systems Manager Fleet Manager è un'esperienza di interfaccia utente unificata che ti aiuta a gestire da remoto i nodi in esecuzione su AWS oppure on-premises.

Risorse

Documenti correlati:

- What is Amazon WorkSpaces?
- Cost Optimizer for Amazon WorkSpaces
- Documentazione di Amazon AppStream 2.0
- NICE DCV

Video correlati:

- Managing cost for Amazon WorkSpaces on AWS

SUS02-BP06 Implementare il buffering o il throttling per appiattire la curva di domanda

Il buffering e la limitazione (della larghezza di banda della rete) riducono la curva delle richieste e la capacità allocata per il tuo carico di lavoro.

Anti-pattern comuni:

- Elabori immediatamente le richieste del client, anche se non è necessario.
- Non analizzi i requisiti relativi alle richieste dei clienti.

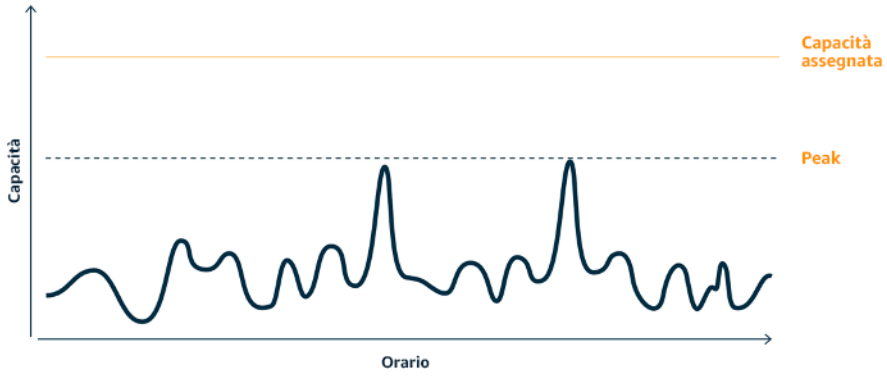
Vantaggi dell'adozione di questa best practice: riduzione della curva della domanda in modo da diminuire la capacità allocata richiesta per il carico di lavoro Ridurre la capacità allocata significa ridurre il consumo di energia e contenere l'impatto ambientale.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

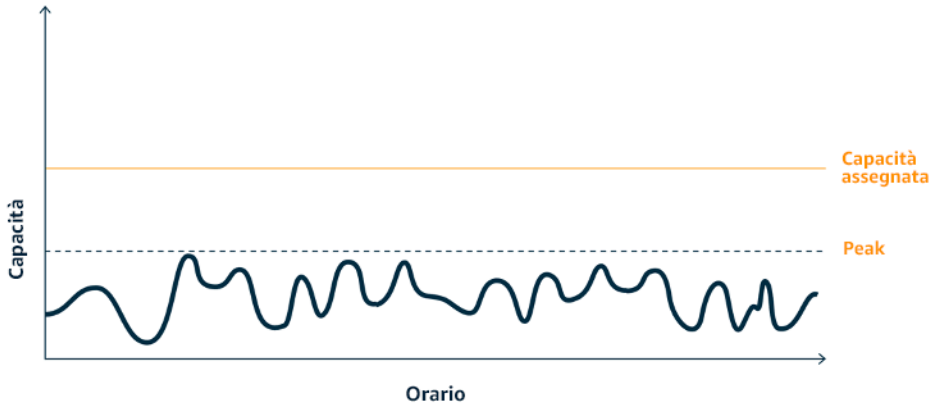
Diminuire la curva della domanda del carico di lavoro può aiutarti a ridurre la capacità allocata di un carico di lavoro, oltre al suo impatto sull'ambiente. Supponiamo che un carico di lavoro abbia la curva della domanda mostrata nella figura qui sotto. Questo carico di lavoro presenta due picchi

e per gestire tali picchi viene eseguito il provisioning della capacità di risorse mostrata dalla linea arancione. Le risorse e l'energia utilizzate per questo carico di lavoro non sono indicate nell'area sotto la curva della domanda, ma nell'area sotto la linea della capacità fornita, poiché per gestire questi due picchi è necessario eseguire il provisioning di tale capacità.



Curva di domanda con due picchi distinti che richiedono un'elevata capacità allocata.

Puoi usare il buffering o la limitazione (della larghezza di banda della rete) per modificare la curva della domanda e appianare i picchi, con conseguente diminuzione della capacità allocata e consumo inferiore di energia. Implementa la limitazione (della larghezza di banda della rete) quando i client eseguono nuovi tentativi. Implementa il buffering per archiviare la richiesta e rinviare l'elaborazione a un secondo momento.



Effetto della limitazione (della larghezza di banda della rete) sulla curva della domanda e sulla capacità allocata.

Passaggi dell'implementazione

- Analizza le richieste del client per stabilire come rispondere. Le domande da considerare includono:
  - Questa richiesta può essere elaborata in modo asincrono?
  - Il client ha la possibilità di ripetere i tentativi?
- Se il client ha la possibilità di ripetere i tentativi puoi implementare la limitazione (della larghezza di banda della rete), che indica alla sorgente che, se non è in grado di soddisfare la richiesta all'ora corrente, dovrebbe riprovare più tardi.
  - Puoi utilizzare Amazon API Gateway per implementare il throttling.
- Per i client che non possono eseguire altri tentativi, è necessario implementare un buffer per ridurre i picchi della curva della domanda. Il buffering rinvia l'elaborazione delle richieste, consentendo alle applicazioni eseguite a velocità diverse di comunicare in modo efficace. Un approccio basato sul buffering impiega una coda o un flusso per l'accettazione dei messaggi dai produttori. I messaggi vengono letti ed elaborati dai consumatori e ciò consente ai messaggi di essere eseguiti alla velocità che soddisfa i requisiti aziendali del consumatore stesso.
  - Amazon Simple Queue Service (AmazonSQS) è un servizio gestito che fornisce code che consentono a un singolo consumatore di leggere singoli messaggi.
  - Amazon Kinesis offre un flusso che consente a più consumatori di leggere gli stessi messaggi.
- Analizza la domanda complessiva, la velocità di modifica e il tempo di risposta richiesto per determinare le dimensioni della limitazione (della larghezza di banda della rete) o del buffer richiesto.

Risorse

Documenti correlati:

- Guida introduttiva ad Amazon SQS
- Application integration Using Queues and Messages
- Gestione e monitoraggio della API limitazione dei carichi di lavoro
- Limitazione su larga scala di un sistema multi-tenant su più livelli utilizzando Gateway REST API API
- Application integration Using Queues and Messages

Video correlati:

- AWS re:Invent 2022 - Modelli di integrazione delle applicazioni per microservizi

- AWS re:Invent 2023 - Risparmio intelligente: strategie di ottimizzazione dei costi di Amazon EC2
- AWS re:Invent 2023 - Modelli di integrazione avanzati e compromessi per sistemi scarsamente accoppiati

Software e architettura

Domanda

- SUS 3. In che modo sfrutti i modelli di software e architetture per sostenere i tuoi obiettivi di sostenibilità?

SUS 3. In che modo sfrutti i modelli di software e architetture per sostenere i tuoi obiettivi di sostenibilità?

Implementa modelli per eseguire lo smoothing del carico e garantire un utilizzo elevato e coerente delle risorse implementate per ridurre al minimo il loro consumo. In seguito alle modifiche nei comportamenti degli utenti nel tempo, alcuni componenti potrebbero diventare inattivi per mancanza di utilizzo. Rivedi modelli e architetture per consolidare i componenti sottoutilizzati e aumentare l'uso complessivo. Ritira i componenti non più necessari. Analizza le prestazioni dei componenti dei tuoi carichi di lavoro e ottimizza quelli che usano la maggior quantità di risorse. Identifica i dispositivi che i clienti utilizzano per accedere ai servizi e implementa modelli in grado di ridurre al minimo la necessità di aggiornamenti dei dispositivi.

Best practice

- SUS03-BP01 Ottimizzazione del software e delle architetture per processi asincroni e pianificati
- SUS03-BP02 Rimozione o rifattorizzazione dei componenti dei carichi di lavoro con un utilizzo ridotto o assente
- SUS03-BP03 Ottimizzazione delle aree di codice che consumano la maggior parte del tempo o delle risorse
- SUS03-BP04 Ottimizzazione dell'impatto su dispositivi e apparecchiature
- SUS03-BP05 Uso dei modelli e le architetture software che meglio supportano l'accesso ai dati e i modelli di archiviazione

SUS03-BP01 Ottimizzazione del software e delle architetture per processi asincroni e pianificati

Utilizza modelli efficienti di software e di architettura, come quelli basati sulle code, per mantenere un utilizzo elevato e costante delle risorse distribuite.

Anti-pattern comuni:

- Provisioning di risorse in eccedenza per il carico di lavoro in cloud con lo scopo di far fronte a picchi di domanda imprevisti.
- Architettura non in grado di disaccoppiare i mittenti e i ricevitori di messaggi asincroni mediante un componente di messaggistica.

Vantaggi dell'adozione di questa best practice:

- Modelli efficienti di software e architettura riducono al minimo le risorse inutilizzate nel carico di lavoro e migliorano l'efficienza complessiva.
- È possibile scalare le risorse dedicate all'elaborazione indipendentemente dalla ricezione di messaggi asincroni.
- Grazie a un componente di messaggistica, i requisiti di disponibilità si attenuano e possono essere soddisfatti con un numero inferiore di risorse.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Utilizza modelli di architettura efficienti come l'architettura basata su eventi così da ottenere un utilizzo uniforme dei componenti, oltre alla riduzione al minimo del provisioning eccessivo nel carico di lavoro. L'utilizzo di modelli architetturali efficienti riduce al minimo le risorse inattive a causa del mancato utilizzo dovuto alle variazioni della domanda nel tempo.

Comprendi i requisiti dei componenti del carico di lavoro e adotta modelli di architettura che aumentino l'utilizzo complessivo delle risorse. Ritira i componenti non più necessari.

Passaggi dell'implementazione

- Analizza le esigenze del tuo carico di lavoro per determinare come rispondere a tali richieste.
- Per le richieste o i processi che non necessitano di risposte sincrone, utilizza architetture basate su code e worker a dimensionamento automatico per massimizzare l'utilizzo. Ecco alcuni esempi in cui potresti prendere in considerazione un'architettura basata sulle code:

Meccanismo di accodamento	Descrizione
Code di processi AWS Batch	I processi AWS Batch vengono inviati a una coda di processi, dove risiedono fino a quando non è possibile pianificare la loro esecuzione in un ambiente di calcolo.
Amazon Simple Queue Service e istanze spot Amazon EC2	Abbina Amazon SQS e istanze spot per realizzare un'architettura efficiente e in grado di tollerare i malfunzionamenti.

- Per le richieste o i processi che possono essere elaborati in qualsiasi momento, ottieni una maggiore efficienza utilizzando i meccanismi di pianificazione dell'elaborazione delle attività in blocco. Ecco alcuni esempi di meccanismi di pianificazione su AWS:

Meccanismo di pianificazione	Descrizione
Pianificatore Amazon EventBridge	Una funzione di Amazon EventBridge per la creazione, esecuzione e gestione di attività pianificate su vasta scala.
Pianificazione basata sul tempo di AWS Glue	Definisci una pianificazione in base al tempo per crawler e processi in AWS Glue.
Processi pianificati di Amazon Elastic Container Service (Amazon ECS)	Amazon ECS supporta la creazione di processi pianificati. I processi pianificati utilizzano le regole di Amazon EventBridge per eseguire processi in base a una pianifica zione o in risposta a un evento EventBridge.
Instance Scheduler	Configura le pianificazioni di avvio e arresto delle istanze di Amazon EC2 e Amazon Relational Database Service.

- Se nella tua architettura utilizzi meccanismi di polling e webhook, sostituiscili con eventi. Utilizza architetture basate sugli eventi per la creazione di carichi di lavoro a elevata efficienza.
- Sfrutta la tecnologia serverless di AWS per eliminare infrastrutture con provisioning eccessivo.

- Dimensiona in modo appropriato i singoli componenti dell'architettura per evitare la presenza di risorse inattive in attesa di input.
  - Puoi sfruttare i suggerimenti per il ridimensionamento corretto in AWS Cost Explorer o AWS Compute Optimizer per individuare le opportunità di dimensionamento corretto.
  - Per ulteriori dettagli, consulta Ridimensionamento corretto: provisioning delle istanze per soddisfare i carichi di lavoro.

Risorse

Documenti correlati:

- What is Amazon Simple Queue Service?
- What is Amazon MQ?
- Scaling based on Amazon SQS
- Cos'è AWS Step Functions?
- Cos'è AWS Lambda?
- Using AWS Lambda with Amazon SQS
- What is Amazon EventBridge?
- Managing Asynchronous Workflows with a REST API

Video correlati:

- AWS re:Invent 2023 - Navigating the journey to serverless event-driven architecture
- AWS re:Invent 2023 - Using serverless for event-driven architecture & domain-driven design
- AWS re:Invent 2023 - Advanced event-driven patterns with Amazon EventBridge
- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- Asynchronous Message Patterns | AWS Events

Esempi correlati:

- Event-driven architecture with AWS Graviton Processors and Amazon EC2 Spot Instances

SUS03-BP02 Rimozione o rifattorizzazione dei componenti dei carichi di lavoro con un utilizzo ridotto o assente

Elimina i componenti non utilizzati e non più necessari e procedi a rifattorizzare quelli con scarso utilizzo per limitare lo spreco di risorse nel tuo carico di lavoro.

Anti-pattern comuni:

- Non verifichi con regolarità il livello di utilizzo dei singoli componenti del tuo carico di lavoro.
- Non segui i consigli ricevuti dagli strumenti di ridimensionamento corretto AWS, ad esempio AWS Compute Optimizer.

Vantaggi dell'adozione di questa best practice: riduzione al minimo degli sprechi e miglioramento dell'efficienza complessiva del carico di lavoro cloud grazie alla rimozione dei componenti non utilizzati.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

I componenti inutilizzati o sottoutilizzati in un carico di lavoro cloud consumano risorse di elaborazione, archiviazione o rete non necessarie. Rimuovi o rifattorizza questi componenti per ridurre direttamente gli sprechi e migliorare l'efficienza complessiva di un carico di lavoro cloud. Si tratta di un processo di miglioramento iterativo che può essere attivato da cambiamenti della domanda o dal rilascio di un nuovo servizio cloud. Ad esempio, una riduzione significativa del runtime delle funzioni di AWS Lambda può indicare la necessità di diminuire la dimensione della memoria Inoltre, quando AWS rilascia nuovi servizi e funzionalità, è possibile che i servizi ottimali e l'architettura per il carico di lavoro cambino.

Monitora continuamente l'attività del carico di lavoro e cerca le opportunità per migliorare il livello di utilizzo dei singoli componenti. Eliminando i componenti inattivi ed eseguendo attività di ridimensionamento corretto, soddisfi i requisiti aziendali con il numero minimo di risorse cloud.

Passaggi dell'implementazione

- Esegui l'inventario delle risorse AWS: crea un inventario delle tue risorse AWS. In AWS, puoi attivare Esploratore di risorse AWS per esaminare e organizzare le tue risorse AWS. Per ulteriori dettagli, guarda AWS re:Invent 2022 - How to manage resources and applications at scale on AWS.

- Monitora l'utilizzo: monitora e acquisisci metriche di utilizzo per i componenti critici del tuo carico di lavoro (come l'utilizzo di CPU e memoria o il throughput di rete nelle metriche di Amazon CloudWatch).
- Identifica i componenti inutilizzati: individua i componenti inutilizzati o sottoutilizzati nell'architettura.
  - In merito ai carichi di lavoro stabili, controlla gli strumenti di ridimensionamento corretto AWS, come AWS Compute Optimizer, a intervalli regolari, così da individuare i componenti inattivi, inutilizzati o sottoutilizzati.
  - Per carichi di lavoro effimeri, valuta metriche di utilizzo per identificare componenti inattivi, inutilizzati o sottoutilizzati.
- Rimuovi i componenti inutilizzati: ritira componenti e risorse associate (come le immagini Amazon ECR) che non sono più necessari.
  - Automated Cleanup of Unused Images in Amazon ECR
  - Delete unused Amazon Elastic Block Store (Amazon EBS) volumes by using AWS Config and AWS Systems Manager
- Rifattorizza i componenti sottoutilizzati: rifattorizza o consolida i componenti sottoutilizzati con altre risorse per promuovere un utilizzo efficiente. Ad esempio, puoi allocare più database di dimensioni ridotte su una singola istanza di database Amazon RDS anziché eseguire database su singole istanze sottoutilizzate.
- Valuta i miglioramenti: scopri le risorse allocate in provisioning dal tuo carico di lavoro per completare un'unità di lavoro. Utilizza queste informazioni per valutare i miglioramenti ottenuti rimuovendo o rifattorizzando i componenti.
  - Measure and track cloud efficiency with sustainability proxy metrics, Part I: What are proxy metrics?
  - Measure and track cloud efficiency with sustainability proxy metrics, Part II: Establish a metrics pipeline

Risorse

Documenti correlati:

- AWS Trusted Advisor
- What is Amazon CloudWatch?
- Ridimensionamento corretto: provisioning delle istanze per soddisfare i carichi di lavoro
- Optimizing your cost with Rightsizing Recommendations

Video correlati:

- AWS re:Invent 2023 - Capacity, availability, cost efficiency: Pick three

SUS03-BP03 Ottimizzazione delle aree di codice che consumano la maggior parte del tempo o delle risorse

Ottimizza il codice eseguito all'interno di diversi componenti della tua architettura per ridurre l'utilizzo delle risorse e massimizzare al tempo stesso le prestazioni.

Anti-pattern comuni:

- Ignori l'ottimizzazione del codice per l'utilizzo delle risorse.
- In genere, rispondi ai problemi di performance aumentando le risorse.
- La revisione del codice e il processo di sviluppo non monitorano le modifiche a livello di performance.

Vantaggi dell'adozione di questa best practice: riduzione al minimo delle risorse utilizzate e ottimizzazione delle prestazioni grazie all'utilizzo di codice efficiente.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

È fondamentale esaminare ogni area funzionale, incluso il codice per un'applicazione ideata nel cloud, per ottimizzare l'uso delle risorse e le performance. Monitora costantemente le performance del tuo carico di lavoro negli ambienti di sviluppo e produzione e identifica le opportunità per migliorare gli snippet di codice che comportano un utilizzo particolarmente elevato delle risorse. Adotta un processo di revisione con cadenza regolare per identificare i bug o gli anti-pattern all'interno del codice che utilizzano le risorse in modo non efficiente. Sfrutta algoritmi semplici ed efficienti che hanno gli stessi risultati per il tuo caso d'uso.

Passaggi dell'implementazione

- Utilizza un linguaggio di programmazione efficiente: usa un sistema operativo e un linguaggio di programmazione efficienti per il carico di lavoro. Per dettagli sui linguaggi di programmazione efficienti dal punto di vista delle risorse (incluso Rust), consulta Sustainability with Rust.

- Usa un assistente per la scrittura di codice basato sull'IA: valuta la possibilità di utilizzare un assistente per la scrittura di codice basato sull'IA, come Amazon Q Developer, per una scrittura efficiente del codice.
- Automatizza le revisioni del codice: mentre sviluppi i tuoi carichi di lavoro, adotta un processo di revisione del codice automatizzato, per migliorar la qualità e identificare bug e anti-pattern.
  - Automate code reviews with Amazon CodeGuru Reviewer
  - Detecting concurrency bugs with Amazon CodeGuru
  - Raising code quality for Python applications using Amazon CodeGuru
- Usa un profiler di codice: utilizza un profiler di codice per identificare le aree di codice che utilizzano la maggior parte del tempo o delle risorse e trasformale in obiettivi di ottimizzazione.
  - Reducing your organization's carbon footprint with Amazon CodeGuru Profiler
  - Understanding memory usage in your Java application with Amazon CodeGuru Profiler
  - Improving customer experience and reducing cost with Amazon CodeGuru Profiler
- Monitora e ottimizza: utilizza risorse di monitoraggio continuo per individuare i componenti con requisiti elevati in termini di risorse o con una configurazione non ottimale.
  - Sostituisci gli algoritmi a uso intensivo di elaborazioni con una versione più semplice ed efficiente che produce gli stessi risultati.
  - Rimuovi il codice non necessario, come quello relativo all'ordinamento e alla formattazione.
- Usa la rifattorizzazione o la trasformazione del codice: scopri le funzionalità di trasformazione del codice Amazon Q per l'esecuzione di manutenzione e aggiornamenti delle applicazioni.
  - Upgrade language versions with Amazon Q Code Transformation
  - AWS re:Invent 2023 - Automate app upgrades & maintenance using Amazon Q Code Transformation

Risorse

Documenti correlati:

- What is Amazon CodeGuru Profiler?
- Istanze FPGA
- SDK AWS su Strumenti per creare su AWS

Video correlati:

- Improve Code Efficiency Using Amazon CodeGuru Profiler
- Automate Code Reviews and Application Performance Recommendations with Amazon CodeGuru

SUS03-BP04 Ottimizzazione dell'impatto su dispositivi e apparecchiature

Individua i dispositivi e le apparecchiature utilizzati nell'architettura e applica le strategie per ridurne l'utilizzo. Questo può ridurre l'impatto ambientale complessivo del tuo carico di lavoro cloud.

Anti-pattern comuni:

- Ignori l'impatto ambientale dei dispositivi utilizzati dai clienti.
- Gestisci e aggiorni manualmente le risorse utilizzate dai clienti.

Vantaggi della definizione di questa best practice: riduzione dell'impatto ambientale complessivo del carico di lavoro sul cloud grazie all'implementazione di modelli e funzionalità software ottimizzati per i dispositivi dei clienti.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Implementare modelli e funzionalità software ottimizzati per i dispositivi dei clienti può ridurre l'impatto ambientale in diversi modi:

- Implementare nuove funzionalità compatibili con le versioni precedenti può ridurre il numero di sostituzioni hardware.
- Ottimizzare un'applicazione per un'esecuzione ottimale sui dispositivi può contribuire a ridurre l'utilizzo di energia ed estendere la durata della relativa batteria (se alimentati in questo modo).
- Ottimizzare un'applicazione per i dispositivi significa anche ridurre il trasferimento dei dati sulla rete.

Conoscere dispositivi e apparecchiature utilizzati nella tua architettura, il loro ciclo di vita atteso e l'impatto della sostituzione di tali componenti. Implementare modelli e funzionalità software in grado di contribuire a ridurre l'uso di energia da parte del dispositivo, la necessità da parte dei clienti di sostituirlo, nonché di eseguire l'aggiornamento manuale.

Passaggi dell'implementazione

- Predisponi un inventario: fai un inventario dei dispositivi usati nella tua architettura. I dispositivi possono essere cellulari, tablet, dispositivi IOT, illuminazione smart o persino dispositivi smart in una fabbrica.
- Utilizza dispositivi a basso consumo energetico: prendi in considerazione l'uso dispositivi a basso consumo energetico nella tua architettura. Utilizza le configurazioni di gestione dell'alimentazione sui dispositivi per accedere alla modalità di risparmio energetico quando non sono in uso.
- Esegui applicazioni efficienti: ottimizza l'applicazione in esecuzione sui dispositivi.
  - Usa strategie come l'esecuzione di attività in background per ridurre l'uso di energia.
  - Prendi in considerazione latenza e larghezza di banda della rete durante la creazione di payload e implementa funzionalità che consentano alle tue applicazioni di funzionare in modo ottimale anche in presenza di una larghezza di banda ridotta e di link ad alta latenza.
  - Converti payload e file in formati ottimizzati richiesti dai dispositivi. Ad esempio, puoi usare Amazon Elastic Transcoder o AWS Elemental MediaConvert per convertire file multimediali digitali di alta qualità di grandi dimensioni nei formati utilizzati dagli utenti per la riproduzione su dispositivi mobili, tablet, browser Web e televisioni connesse.
  - Esegui attività a elevata intensità di calcolo lato server (come il rendering delle immagini) oppure usa lo streaming delle applicazioni per migliorare l'esperienza utente sui dispositivi meno recenti.
  - Esegui la segmentazione e la paginazione dell'output, soprattutto per le sessioni interattive, al fine di gestire i payload e limitare i requisiti di archiviazione in locale.
- Coinvolgi i fornitori: collabora con i fornitori dei dispositivi che utilizzano materiali sostenibili e garantiscono trasparenza circa le loro catene di approvvigionamento e certificazioni ambientali.
- Utilizza aggiornamenti via etere (OTA): usa un meccanismo via etere (OTA) automatizzato per implementare gli aggiornamenti in uno o più dispositivi.
  - Per aggiornare le applicazioni mobili, puoi utilizzare una pipeline CI/CD.
  - Puoi usare AWS IoT Device Management per gestire in remoto i dispositivi connessi su larga scala.
- Usa device farm gestite: per testare nuove funzionalità e aggiornamenti, usa device farm gestite con set di hardware rappresentativi e itera lo sviluppo per ottimizzare i dispositivi supportati. Per ulteriori dettagli, consultare SUS06-BP05 Utilizzo di device farm gestite per i test.
- Continua a monitorare e apportare miglioramenti: monitora il consumo energetico dei dispositivi per identificare le aree di miglioramento. Utilizza le nuove tecnologie o best practice per migliorare l'impatto ambientale di tali dispositivi.

Risorse

Documenti correlati:

- Cos'è AWS Device Farm?
- Documentazione sulle applicazioni WorkSpaces
- NICE DCV
- Tutorial OTA per aggiornare i firmware sui dispositivi che eseguono FreerTOS
- Optimizing Your IoT Devices for Environmental Sustainability

Video correlati:

- AWS re:Invent 2023 - Improve your mobile and web app quality using AWS Device Farm

SUS03-BP05 Uso dei modelli e le architetture software che meglio supportano l'accesso ai dati e i modelli di archiviazione

Scopri come i dati vengono utilizzati all'interno del tuo carico di lavoro, consumati dagli utenti, trasferiti e archiviati. Usa architetture e modelli software in grado di supportare al meglio l'accesso ai dati e l'archiviazione per ridurre le risorse di elaborazione, rete e storage richieste dal carico di lavoro.

Anti-pattern comuni:

- Ritieni che tutti i carichi di lavoro abbiano modelli di accesso e archiviazione di dati simili.
- Utilizzi un solo livello di archiviazione, presupponendo che tutti i carichi di lavoro rientrino in tale livello.
- Ritieni che gli schemi di accesso ai dati rimarranno coerenti nel tempo.
- La tua architettura supporta una potenziale espansione elevata dell'accesso ai dati, con conseguente inattività delle risorse per la maggior parte del tempo.

Vantaggi dell'adozione di questa best practice: riduzione della complessità dello sviluppo e aumento dell'utilizzo complessivo grazie alla selezione e all'ottimizzazione dell'architettura in base ai modelli di accesso ai dati e di archiviazione. Capire quando utilizzare le tabelle globali, il partizionamento dei dati e la memorizzazione nella cache, ti aiuterà a ridurre i costi operativi e a effettuare il dimensionamento in base alle esigenze del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Per migliorare la sostenibilità del carico di lavoro a lungo termine, utilizza modelli di architettura che supportino le caratteristiche di storage e accesso ai dati per il tuo carico di lavoro. Tali modelli ti aiutano a recuperare ed elaborare i dati in modo efficiente. Ad esempio, puoi utilizzare un'architettura dati moderna su AWS con servizi appositamente progettati e ottimizzati per i tuoi specifici casi d'uso di analisi. Questi modelli di architettura consentono un'elaborazione efficiente dei dati e riducono l'utilizzo delle risorse.

Passaggi dell'implementazione

- Comprensione delle caratteristiche dei dati: analizza le caratteristiche dei dati e i modelli di accesso per individuare la configurazione corretta per le tue risorse cloud. Gli aspetti chiave da considerare includono:
  - Tipo di dati: strutturati, semi-strutturati, non strutturati
  - Crescita dei dati: limitata, illimitata
  - Durabilità dei dati: persistenti, effimeri, transitori
  - Schemi di accesso: letture o scritture, frequenza di aggiornamento, con picchi o costante
- Utilizzo di modelli di architettura ottimali: utilizza tipi di architetture che meglio supportino l'accesso ai dati e i modelli di archiviazione.
  - Patterns for enabling data persistence
  - Let's Architect! Modern data architectures
  - Databases on AWS: The Right Tool for the Right Job
- Utilizzo di servizi appositamente progettati: utilizza tecnologie che sono adatte allo specifico caso d'uso.
  - Sfrutta le tecnologie che lavorano in modo nativo con i dati compressi.
    - Athena Compression Support file formats
    - Format Options for ETL Inputs and Outputs in AWS Glue
    - Loading compressed data files from Amazon S3 with Amazon Redshift
  - Sfrutta servizi di analisi appositamente creati per l'elaborazione dei dati nella tua architettura. Per informazioni dettagliate sui servizi di analisi AWS appositamente creati, guarda AWS re:Invent 2022 - Building modern data architectures on AWS.
  - Utilizza il motore del database che meglio supporta il modello di query dominante. Gestisci gli indici di database per un'esecuzione efficiente delle query. Per ulteriori informazioni, consulta Database su AWS e guarda AWS re:Invent 2022 - Modernize apps with purpose-built databases.

- Riduzione al minimo dei trasferimenti di dati: seleziona protocolli di rete che riducano la quantità di capacità di rete utilizzata dalla tua architettura.

Risorse

Documenti correlati:

- COPY from columnar data formats with Amazon Redshift
- Converting Your Input Record Format in Firehose
- Migliora le prestazioni delle query su Amazon Athena con una conversione ai formati in colonne
- Monitoring DB load with Performance Insights on Amazon Aurora
- Monitoring DB load with Performance Insights on Amazon RDS
- Classe di archiviazione del Piano intelligente Amazon S
- Build a CQRS event store with Amazon DynamoDB

Video correlati:

- AWS re:Invent 2022 - Building data mesh architectures on AWS
- AWS re:Invent 2023 - Deep dive into Amazon Aurora and its innovations
- AWS re:Invent 2023 - Improve Amazon EBS efficiency and be more cost-efficient
- AWS re:Invent 2023 - Optimizing storage price and performance with Amazon S3
- AWS re:Invent 2023 - Building and optimizing a data lake on Amazon S3
- AWS re:Invent 2023 - Advanced event-driven patterns with Amazon EventBridge

Esempi correlati:

- AWS Purpose Built Databases Workshop
- AWS Modern Data Architecture Immersion Day
- Build a Data Mesh on AWS

Dati

Domanda

- SUS 4. Come si può usufruire delle policy e dei modelli di gestione dei dati per supportare gli obiettivi di sostenibilità?

SUS 4. Come si può usufruire delle policy e dei modelli di gestione dei dati per supportare gli obiettivi di sostenibilità?

Implementa procedure di gestione dei dati per ridurre l'archiviazione allocata richiesta per supportare il carico di lavoro e le risorse necessarie per l'uso correlato. Analizza i tuoi dati e usa tecnologie e configurazioni di archiviazione che supportano più efficacemente il valore aziendale dei dati e il modo in cui vengono utilizzati. Esegui il ciclo di vita dei dati su un'archiviazione più efficiente e meno performante al diminuire dei requisiti ed elimina i dati che non sono più necessari.

Best practice

- SUS04-BP01 Implementare una politica di classificazione dei dati
- SUS04-BP02 Utilizzo di tecnologie che supportano l'accesso ai dati e i modelli di archiviazione
- SUS04-BP03 Utilizzo delle policy per gestire il ciclo di vita dei set di dati
- SUS04-BP04 Usa l'elasticità e l'automazione per espandere lo storage a blocchi o il file system
- SUS04-BP05 Eliminazione dei dati ridondanti o non necessari
- SUS04-BP06 Utilizzo di file system condivisi o archiviazione per accedere a dati comuni
- SUS04-BP07 Riduzione al minimo dello spostamento di dati tra reti
- SUS04-BP08 Backup dei dati solo quando sono difficili da ricreare

SUS04-BP01 Implementare una politica di classificazione dei dati

Classifica i dati per capire le criticità rispetto ai risultati aziendali e scegli il livello di archiviazione ad alta efficienza corretto per le tue informazioni.

Anti-pattern comuni:

- Non identifichi asset di dati con caratteristiche simili (come sensibilità, criticità aziendale o requisiti normativi) che vengono elaborati o archiviati.
- Non hai implementato un catalogo di dati per eseguire l'inventario dei tuoi asset.

Vantaggi dell'adozione di questa best practice: determinazione del livello di archiviazione dei dati più efficiente dal punto di vista energetico grazie all'implementazione di una policy di classificazione dei dati.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

La classificazione dei dati comporta l'identificazione dei tipi di dati elaborati e archiviati in un sistema informativo di proprietà o gestito da un'organizzazione. Inoltre, è necessario stabilire la criticità dei dati e il probabile impatto di una compromissione, perdita o uso improprio dei dati.

Implementare la policy di classificazione dei dati partendo dall'uso contestuale dei dati e creando uno schema di categorizzazione che tenga conto del livello di criticità di un determinato set di dati per le operazioni dell'organizzazione.

Passaggi dell'implementazione

- Esegui l'inventario dei dati: redigi l'inventario dei vari tipi di dati esistenti per il carico di lavoro.
- Raggruppa i dati: determina la criticità, la riservatezza, l'integrità e la disponibilità dei dati in base al rischio per l'organizzazione. Utilizza questi requisiti per raggruppare i dati in uno dei livelli di classificazione dei dati adottati. Ad esempio, consulta Quattro semplici passaggi per classificare i dati e proteggere la tua startup.
- Definisci livelli di classificazione dei dati e policy: per ciascun gruppo di dati, definisci il livello di classificazione dei dati (ad esempio, pubblico o riservato) e le policy di gestione. Applica ai dati i tag adeguati. Per maggiori dettagli sulle categorie di classificazione dei dati, consulta il whitepaper sulla classificazione dei dati.
- Rivedi periodicamente: esamina e controlla periodicamente l'ambiente per verificare la presenza di dati senza tag e non classificati. Usa l'automazione per identificare questi dati, classificandoli e applicando i tag in modo appropriato. Ad esempio, consulta Data Catalog and crawlers in AWS Glue.
- Crea un catalogo dati: definisci un catalogo dati con funzionalità di audit e governance
- Documenta: crea documenti relativi a policy di classificazione dei dati e procedure di gestione per ciascuna classe di dati.

Risorse

Documenti correlati:

- Leveraging Cloud AWS to Support Data Classification
- Politiche di tag da AWS Organizations

Video correlati:

- AWS re:Invent 2022 - Promuovere l'agilità con la governance dei dati attiva AWS
- AWS re:Invent 2023 - Protezione e resilienza dei dati con storage AWS

SUS04-BP02 Utilizzo di tecnologie che supportano l'accesso ai dati e i modelli di archiviazione

Usa tecnologie di archiviazione in grado di supportare al meglio il modo in cui viene effettuato l'accesso ai dati e come vengono archiviati per ridurre la quantità di risorse allocate e supportare al tempo stesso il tuo carico di lavoro.

Anti-pattern comuni:

- Ritieni che tutti i carichi di lavoro abbiano modelli di accesso e archiviazione di dati simili.
- Utilizzi un solo livello di archiviazione, presupponendo che tutti i carichi di lavoro rientrino in tale livello.
- Ritieni che gli schemi di accesso ai dati rimarranno coerenti nel tempo.

Vantaggi dell'adozione di questa best practice: selezionare e ottimizzare le tecnologie di archiviazione in base all'accesso ai dati e ai modelli di archiviazione ti consentirà di ridurre le risorse cloud richieste per soddisfare le tue esigenze aziendali e migliorare l'efficienza generale del tuo carico di lavoro cloud.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all’implementazione

Seleziona la soluzione di archiviazione più adatta ai tuoi modelli di accesso. In alternativa, puoi modificarli affinché siano in linea con la soluzione di archiviazione, allo scopo di ottimizzare l'efficienza delle prestazioni.

Passaggi dell'implementazione

- Esamina le caratteristiche dei dati e dell'accesso: valuta le caratteristiche dei tuoi dati e il modello di accesso per raccogliere le caratteristiche chiave delle tue esigenze di archiviazione. Gli aspetti chiave da considerare includono:
  - Tipo di dati: strutturati, semi-strutturati, non strutturati
  - Crescita dei dati: limitata, illimitata
  - Durabilità dei dati: persistenti, effimeri, transitori
  - Modelli di accesso: letture o scritture, frequenza, con picchi o costante
- Scegli la giusta tecnologia di archiviazione: migra i dati alla tecnologia di archiviazione appropriata che supporta le caratteristiche dei tuoi dati e il modello di accesso. Ecco alcuni esempi di tecnologie di archiviazione AWS e delle loro caratteristiche chiave:

Tipo	Tecnologia	Caratteristiche chiave
Archiviazione di oggetti	Amazon S3	Un servizio di archiviazione di oggetti con scalabilità illimitata, elevata disponibilità e più opzioni di accessibilità. Il trasferimento di oggetti e il relativo trasferimento da e verso Amazon S3 può utilizzare un servizio, come Transfer Acceleration o Punti di accesso, per supportare la posizione, le esigenze di sicurezza e i modelli di accesso.
Archiviazione	Amazon Glacier	Classe di archiviazione di Amazon S3 creata per l'archiviazione dei dati.
File system condiviso	Amazon Elastic File System (Amazon EFS)	File system montabile a cui è possibile accedere da più tipi di soluzioni di calcolo.

Tipo	Tecnologia	Caratteristiche chiave
		Amazon EFS aumenta e riduce in automatico l'archivi azione e presenta ottimizza zioni in termini di prestazio ni per offrire latenze basse e costanti.
File system condiviso	Amazon FSx	Sviluppato con le più recenti soluzioni di calcolo AWS per supportare i 4 file system più comunemente utilizzati: NetApp ONTAP, OpenZFS, Windows File Server e Lustre. Latenza, throughput e IOPS di Amazon FSx variano a seconda del file system; è necessario considerare attentamente questi elementi quando si deve selezionare il file system in modo conforme ai requisiti dei carichi di lavoro.
Storage a blocchi	Amazon Elastic Block Store (Amazon EBS)	Servizio di storage a blocchi scalabile e a elevate prestazioni progettato per Amazon Elastic Compute Cloud (Amazon EC2). Amazon EBS include storage su SSD per carichi di lavoro transazionali e intensivi dal punto di vista dell'IOPS, oltre a storage su HDD per carichi di lavoro con throughput intensivo.

Tipo	Tecnologia	Caratteristiche chiave
Database relazionale	Amazon Aurora, Amazon RDS, Amazon Redshift	Progettati per supportare le transazioni ACID (atomicità, coerenza, isolamento, durabilità) e per mantenere l'integrità referenziale e una solida coerenza dei dati. Molte applicazioni tradizionali e sistemi Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) ed e-commerce utilizzano database relazionali per archiviare i propri dati.
Database chiave-valore	Amazon DynamoDB	Ottimizzato per schemi di accesso di uso comune, in genere per archiviare e recuperare grandi volumi di dati. Le app Web dal traffico elevato, i sistemi di e-commerce e le applicazioni di videogiochi sono casi d'uso tipici dei database chiave-valore.

- Automatizza l'allocazione dello spazio di archiviazione: per i sistemi di archiviazione con dimensione fissa, come Amazon EBS o Amazon FSx, monitora lo spazio di archiviazione disponibile e automatizza l'allocazione dell'archiviazione al raggiungimento di una soglia. Sfrutta Amazon CloudWatch per raccogliere e analizzare vari parametri per Amazon EBS e Amazon FSx.
- Scegli la classe di archiviazione giusta: scegli la classe di archiviazione opportuna per i tuoi dati.
  - Le classi di archiviazione Amazon S3 possono essere configurate a livello di oggetto. Un singolo bucket può contenere oggetti archiviati per tutte le classi di archiviazione.
  - Puoi utilizzare le policy del ciclo di vita Amazon S3 per passare automaticamente gli oggetti tra le classi di archiviazione oppure rimuovere i dati senza modifiche all'applicazione. In generale,

devi raggiungere un equilibrio tra efficienza delle risorse, latenza di accesso e affidabilità, quando consideri questi meccanismi di storage.

Risorse

Documenti correlati:

- Amazon EBS volume types
- Archivio dell'istanza Amazon EC2
- Amazon S3 Intelligent-Tiering
- Amazon EBS I/O Characteristics
- Using Amazon S3 storage classes
- Cos'è Amazon Glacier?

Video correlati:

- AWS re:Invent 2023 - Improve Amazon EBS efficiency and be more cost-efficient
- AWS re:Invent 2023 - Optimizing storage price and performance with Amazon S3
- AWS re:Invent 2023 - Building and optimizing a data lake on Amazon S3
- AWS re:Invent 2022 - Building modern data architectures on AWS
- AWS re:Invent 2022 - Modernize apps with purpose-built databases
- AWS re:Invent 2022 - Building data mesh architectures on AWS
- AWS re:Invent 2023 - Deep dive into Amazon Aurora and its innovations
- AWS re:Invent 2023 - Advanced data modeling with Amazon DynamoDB

Esempi correlati:

- Amazon S3 Examples
- AWS Purpose Built Databases Workshop
- Databases for Developers
- AWS Modern Data Architecture Immersion Day
- Build a Data Mesh on AWS

SUS04-BP03 Utilizzo delle policy per gestire il ciclo di vita dei set di dati

Gestisci il ciclo di vita di tutti i tuoi dati e applica in automatico le cancellazioni per ridurre i requisiti totali di archiviazione del tuo carico di lavoro.

Anti-pattern comuni:

- Cancellazione manuale dei dati.
- Conservazione di tutti i dati del carico di lavoro.
- Mancato spostamento dei dati su livelli di archiviazione più efficienti dal punto di vista energetico in base ai requisiti di conservazione e accesso.

Vantaggi dell'adozione di questa best practice: l'utilizzo delle policy per il ciclo di vita dei dati garantisce un accesso e una conservazione efficienti dei dati in un carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

i set di dati presentano solitamente requisiti di conservazione e accesso che cambiano durante il loro ciclo di vita. Ad esempio, l'applicazione potrebbe avere bisogno di accedere frequentemente ad alcuni set di dati per un periodo di tempo limitato. In seguito, questi set di dati vengono consultati di rado. Per migliorare l'efficienza dell'archiviazione e dell'elaborazione dei dati nel tempo, implementa le policy per il ciclo di vita (regole che definiscono il modo in cui i dati vengono gestiti nel tempo).

Con le regole di configurazione del ciclo di vita, è possibile indicare al servizio di archiviazione di trasferire un set di dati a livelli di archiviazione più efficienti dal punto di vista energetico, di archiviarlo o di eliminarlo. Questa pratica riduce al minimo le operazioni di archiviazione e recupero attive dei dati, con conseguente riduzione del consumo energetico. Inoltre, pratiche come l'archiviazione o l'eliminazione di dati obsoleti favoriscono la conformità normativa e la governance dei dati.

Passaggi dell'implementazione

- Utilizza la classificazione di dati: classifica i set di dati nel tuo carico di lavoro.
- Definisci le procedure di gestione: definisci le procedure di gestione per ogni classe di dati.
- Abilita l'automatizzazione: imposta policy automatizzate per il ciclo di vita affinché vengano applicate le regole correlate. Ecco alcuni esempi di come impostare policy automatizzate per il ciclo di vita di diversi servizi di archiviazione di AWS:

Servizio di storage	Come impostare policy automatizzate per il ciclo di vita
Amazon S3	Puoi utilizzare il ciclo di vita Amazon S3 per gestire gli oggetti durante il loro ciclo di vita. In caso di schemi di accesso sconosciuti, mutevoli o imprevedibili, puoi utilizzare il Piano intelligente Amazon S3, che monitora gli schemi di accesso e sposta in automatico gli oggetti che non hanno fatto registrare accessi a livelli di accessi più economici. Sfrutta i parametri di Amazon S3 Storage Lens per individuare opportunità di ottimizzazione e lacune nella gestione del ciclo di vita.
Amazon Elastic Block Store	È possibile utilizzare Amazon Data Lifecycle Manager per automatizzare la creazione, la conservazione e l'eliminazione degli snapshot Amazon EBS e delle AMI supportate da Amazon EBS.
Amazon Elastic File System	La gestione del ciclo di vita di Amazon EFS gestisce automaticamente lo storage di file a costi contenuti per i file system.
Amazon Elastic Container Registry	Le policy del ciclo di vita di Amazon ECR automatizzano la pulizia delle immagini dei container, facendole scadere in base all'età o al conteggio.
AWS Elemental MediaStore	Puoi creare una policy del ciclo di vita degli oggetti che gestisce la durata di archiviazione degli oggetti nel container MediaStore.

- Elimina i volumi inutilizzati: elimina i volumi inutilizzati, gli snapshot e i dati che hanno superato il periodo di conservazione. Sfrutta le funzionalità di servizio native come Amazon DynamoDB Time To Live o la conservazione dei log di Amazon CloudWatch per l'eliminazione.

- Aggrega e comprimi: aggrega e comprimi i dati quando possibile in base alle regole del ciclo di vita.

Risorse

Documenti correlati:

- Ottimizzazione delle regole del ciclo di vita di Amazon S3 con Amazon S3 Storage Class Analysis
- Evaluating Resources with Regole di AWS Config

Video correlati:

- AWS re:Invent 2021 - Amazon S3 Lifecycle best practices to optimize your storage spend
- AWS re:Invent 2023 - Optimizing storage price and performance with Amazon S3
- Simplify Your Data Lifecycle and Optimize Storage Costs With Amazon S3 Lifecycle
- Reduce Your Storage Costs Using Amazon S3 Storage Lens

SUS04-BP04 Usa l'elasticità e l'automazione per espandere lo storage a blocchi o il file system

Usa l'elasticità e l'automazione per espandere lo storage a blocchi o il file system con l'aumento dei dati per ridurre l'archiviazione allocata.

Anti-pattern comuni:

- Acquisti uno storage a blocchi di grandi dimensioni o un file system per necessità future.
- Il numero di operazioni di input e output al secondo (IOPS) del file system è superiore al numero di operazioni di input e output.
- Non monitori l'utilizzo dei volumi di dati.

Vantaggi dell'adozione di questa best practice: la riduzione del provisioning eccessivo per il sistema di archiviazione riduce le risorse inattive e migliora l'efficienza complessiva del tuo carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Crea storage a blocchi e file system con l'allocazione delle dimensioni, il throughput e la latenza adeguati al tuo carico di lavoro. Usa l'elasticità e l'automazione per espandere lo storage a blocchi

o il file system con l'aumento dei dati per evitare un provisioning eccessivo per questi servizi di archiviazione.

Passaggi dell'implementazione

- Per lo storage a dimensione fissa come Amazon EBS, verifica di monitorare la quantità di storage utilizzata rispetto alla dimensione complessiva dello storage e crea l'automazione, se possibile, per aumentare le dimensioni dello storage quando si raggiunge una soglia.
- Utilizza volumi elastici e servizi di dati a blocchi gestiti per automatizzare l'allocazione di archivi aggiuntivi man mano che i dati persistenti aumentano. Ad esempio, puoi utilizzare Amazon EBS Elastic Volumes per modificare la dimensione del volume, il tipo di volume o regolare le prestazioni dei tuoi EBS volumi Amazon.
- Scegli la classe di archiviazione corretta, le performance e il throughput per il tuo file system per rispondere alle esigenze della tua azienda, senza eccedere.
  - EFSPrestazioni di Amazon
  - Prestazioni dei EBS volumi Amazon su istanze Linux
- Imposta i livelli target di utilizzo per i volumi di dati e ridimensiona i volumi al di fuori degli intervalli previsti.
- Dimensiona i volumi di sola lettura per adattarli ai dati.
- Migra i dati su archivi oggetti per evitare il provisioning di capacità eccessive da dimensioni di volumi fisse su archiviazioni a blocchi.
- Esamina regolarmente i volumi elastici e i file system per terminare i volumi inattivi e ridurre i volumi con un provisioning eccessivo per adattarli alla dimensione corrente dei dati.

Risorse

Documenti correlati:

- Estendi il file system dopo il ridimensionamento di un volume EBS
- Modifica un volume utilizzando Amazon EBS Elastic Volumes
- Documentazione FSx Amazon
- What is Amazon Elastic File System?

Video correlati:

- Approfondimento su Amazon EBS Elastic Volumes

- Strategie di ottimizzazione di Amazon EBS e Snapshot per migliori prestazioni e risparmi sui costi
- Ottimizzazione di Amazon in termini EFS di costi e prestazioni, utilizzando le best practice

SUS04-BP05 Eliminazione dei dati ridondanti o non necessari

Elimina i dati non necessari o ridondanti per ridurre al minimo le risorse di archiviazione necessarie per memorizzare i set di dati.

Anti-pattern comuni:

- Duplicazione dei dati che possono essere facilmente recuperati o ricreati.
- Backup di tutti i dati senza prenderne in considerazione la criticità.
- Cancellazione dei dati eseguita in modo irregolare, in occasione di eventi operativi o non eseguita affatto.
- Archiviazione dei dati in modo ridondante, indipendentemente dall'affidabilità del servizio di archiviazione.
- Attivazione del controllo delle versioni di Amazon S3 senza alcuna giustificazione aziendale.

Vantaggi dell'adozione di questa best practice: riduzione delle dimensioni di archiviazione necessarie per il carico di lavoro e del suo impatto ambientale grazie alla rimozione dei dati non necessari.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Quando rimuovi set di dati non necessari e ridondanti, puoi ridurre i costi di storage e l'impatto ambientale. Questa pratica può anche rendere l'elaborazione più efficiente, poiché le risorse di calcolo elaborano solo dati importanti anziché dati non necessari. Automatizza l'eliminazione dei dati non necessari. Utilizza tecnologie di backup che deduplicano i dati a livello di file e blocco. Sfrutta le funzionalità native dei servizi per la replica e la ridondanza dei dati.

Passaggi dell'implementazione

- Valuta set di dati pubblici: valuta la possibilità di non archiviare i dati utilizzando i set di dati esistenti pubblicamente disponibili in AWS Data Exchange e Open Data su AWS.
- Deduplica i dati: utilizza meccanismi che possano deduplicare i dati a livello di blocco e oggetto. Ecco alcuni esempi di come deduplicare i dati su AWS:

Servizio di storage	Meccanismi di deduplicazione
Amazon S3	Usa AWS Lake Formation FindMatches per trovare i record corrispondenti in un set di dati (compresi quelli senza identificatori) sfruttando il nuovo FindMatches ML Transform.
Amazon FSx	Usa la deduplicazione dei dati su Amazon FSx per Windows.
Volumi e snapshot di Amazon Elastic Block Store	Gli snapshot sono incrementali, ovvero vengono salvati solo i blocchi sul dispositivo che sono cambiati dall'ultimo snapshot.

- Utilizza le policy del ciclo di vita: serviti delle policy del ciclo di vita per automatizzare l'eliminazione dei dati non necessari. Sfrutta funzionalità native come Amazon DynamoDB Time To Live, Amazon S3 Lifecycle o la conservazione dei log di Amazon CloudWatch per l'eliminazione
- Utilizza la virtualizzazione dei dati: utilizza le funzionalità di virtualizzazione dei dati di AWS per mantenere i dati sul sistema di origine ed evitarne la duplicazione.
  - Cloud Native Data Virtualization on AWS
  - Optimize Data Pattern Using Amazon Redshift Data Sharing
- Utilizza il backup incrementale: utilizza una tecnologia di backup in grado di eseguire backup incrementali.
- Utilizza la durabilità nativa: sfrutta la durabilità di Amazon S3 e la replica di Amazon EBS per raggiungere i tuoi obiettivi in termini di persistenza anziché le tecnologie autogestite (come un array ridondante di dischi indipendenti o RAID).
- Utilizza funzionalità efficaci di registrazione dei log: centralizza i log e traccia i dati, deduplica le voci di log identiche e stabilisci meccanismi per ottimizzarne la verbosità quando necessario.
- Utilizza funzionalità efficaci di memorizzazione nella cache: precompila i dati nelle cache solo quando è necessario.
- Definisci il monitoraggio e l'automazione della cache per ridimensionarla in base alle esigenze.
- Rimuovi le versioni obsolete delle risorse: rimuovi le implementazioni e le risorse obsolete dagli archivi di oggetti e dalle cache edge durante la distribuzione di nuove versioni del carico di lavoro.

Risorse

Documenti correlati:

- Change log data retention in CloudWatch Logs
- Deduplicazione dei dati su Amazon FSx per Windows File Server
- Funzionalità di Amazon FSx per ONTAP, inclusa la deduplicazione dei dati
- Invalidating Files on Amazon CloudFront
- Uso di AWS Backup per eseguire il backup e ripristinare i file system di Amazon EFS
- What is Amazon CloudWatch Logs?
- Working with backups on Amazon RDS
- Integrate and deduplicate datasets using AWS Lake Formation

Video correlati:

- Amazon Redshift Data Sharing Use Cases

Esempi correlati:

- Come posso usare Amazon Athena per analizzare i log di accesso al server Amazon S3?

SUS04-BP06 Utilizzo di file system condivisi o archiviazione per accedere a dati comuni

Adotta file system condivisi o l'archiviazione per evitare duplicazioni di dati e abilitare un'infrastruttura più efficiente per il tuo carico di lavoro.

Anti-pattern comuni:

- Esegui il provisioning dell'archiviazione per ogni singolo client.
- Non scolleghi volumi di dati da client inattivi.
- Non fornisci l'accesso allo storage su piattaforme e sistemi.

Vantaggi dell'adozione di questa best practice: condivisione di dati con uno o più utenti senza la necessità di copiarli grazie all'utilizzo di file system o archiviazione condivisi. Questo consente di ridurre le risorse di archiviazione necessarie per il carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Se hai più utenti o applicazioni che accedono agli stessi set di dati, usare una tecnologia di archiviazione condivisa è fondamentale per usare un'infrastruttura efficiente per il tuo carico di lavoro. La tecnologia di archiviazione condivisa offre una posizione centrale per archiviare e gestire set di dati ed evitare la loro duplicazione. Verifica anche la coerenza dei dati su sistemi diversi. Inoltre, la tecnologia di archiviazione condivisa consente un uso più efficiente della potenza di elaborazione, poiché più risorse di calcolo possono accedere ed elaborare i dati allo stesso momento in parallelo.

Acquisisci i dati dai servizi di archiviazione condivisa in base alle necessità e scollega i volumi non utilizzati per liberare le risorse.

Passaggi dell'implementazione

- Usa l'archiviazione condivisa: esegui la migrazione dei dati nell'archiviazione condivisa quando i dati hanno più consumer. Ecco alcuni esempi della tecnologia di archiviazione condivisa su AWS:

Opzione di archiviazione	Quando utilizzare
Amazon EBS Multi-Attach	Amazon EBS Multi-Attach consente di collegare un volume singolo SSD con capacità di IOPS allocata (io1 o io2) a più istanze che si trovano nella stessa zona di disponibilità.
Amazon EFS	Consulta When to Choose Amazon EFS.
Amazon FSx	Consulta Scelta di un file system Amazon FSx.
Amazon S3	Le applicazioni che non richiedono una struttura di file system e sono progettate per lavorare con lo storage degli oggetti possono usare Amazon S3 come soluzione di archiviaz ione degli oggetti a basso costo, durevole e altamente scalabile.

- Acquisisci i dati in base alle necessità: copia o acquisisci i dati solo da file system condivisi in base alle necessità. Ad esempio, puoi creare un file system Amazon FSx per Lustre supportato

da Amazon S3 e caricare solo il sottoinsieme di dati necessario per i processi di elaborazione su Amazon FSx.

- Elimina i dati non necessari: elimina i dati nella modalità corretta per i tuoi modelli di utilizzo come illustrato in SUS04-BP03 Utilizzo delle policy per gestire il ciclo di vita dei set di dati.
- Scollega i client inattivi: scollega i volumi dai client che non li utilizzano attivamente.

Risorse

Documenti correlati:

- Linking your file system to an Amazon S3 bucket
- Using Amazon EFS for AWS Lambda in your serverless applications
- Amazon EFS Intelligent-Tiering Optimizes Costs for Workloads with Changing Access Patterns
- Using Amazon FSx with your on-premises data repository

Video correlati:

- Storage cost optimization with Amazon EFS
- AWS re:Invent 2023 - What's new with AWS file storage
- AWS re:Invent 2023 - File storage for builders and data scientists on Amazon Elastic File System

SUS04-BP07 Riduzione al minimo dello spostamento di dati tra reti

Usa file system condivisi o lo storage a oggetti per accedere ai dati comuni e contenere le risorse di rete totali necessarie per supportare i trasferimenti dei dati per il carico di lavoro.

Anti-pattern comuni:

- Archivi tutti i dati nella stessa Regione AWS, indipendentemente dalla posizione degli utenti.
- Non ottimizzi la dimensione e il formato dei dati prima di trasferirli sulla rete.

Vantaggi dell'adozione di questa best practice: l'ottimizzazione del trasferimento dei dati sulla rete riduce la quantità di risorse di rete totali richieste per il carico di lavoro e diminuisce l'impatto ambientale.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Trasferire i dati all'interno dell'organizzazione significa disporre di risorse di elaborazione, rete e archiviazione. Usa tecniche per ridurre il movimento dei dati e migliorare l'efficienza generale del tuo carico di lavoro.

Passaggi dell'implementazione

- Utilizza la vicinanza: considera la vicinanza ai dati o agli utenti come fattore decisivo per la selezione di una Regione per il tuo carico di lavoro.
- Esegui la partizione dei servizi: esegui la partizione dei servizi utilizzati a livello regionale in modo che i dati specifici della Regione siano conservati nella Regione in cui vengono utilizzati.
- Usa formati di file efficaci: usa formati di file efficaci (come Parquet o ORC) e comprimi i dati prima di spostarli sulla rete.
- Riduci al minimo lo spostamento dei dati: non trasferire dati inutilizzati. Alcuni esempi che possono aiutarti a evitare di spostare dati inutilizzati:
  - Riduci le risposte API solo ai dati pertinenti.
  - Aggrega i dati laddove richiesto (le informazioni a livello di record non sono necessarie).
  - Consulta Well-Architected Lab - Optimize Data Pattern Using Amazon Redshift Data Sharing.
  - Prendi in considerazione la condivisione dei dati tra account in AWS Lake Formation.
- Utilizza servizi edge: utilizza servizi in grado di supportarti nell'esecuzione del codice in posizioni più vicine agli utenti del carico di lavoro.

Servizio	Quando usare
Lambda@Edge	Da utilizzare per operazioni a uso intensivo di risorse di calcolo eseguite quando gli oggetti non si trovano nella cache.
Funzioni CloudFront	Da utilizzare per casi d'uso semplici, ad esempio manipolazioni di risposte o richieste HTTP(s) che possono essere avviate da funzioni di breve durata.

Servizio	Quando usare
AWS IoT Greengrass	Eseguire la memorizzazione nella cache di risorse di calcolo, messaggistica e dati per i dispositivi connessi.

Risorse

Documenti correlati:

- Optimizing your AWS Infrastructure for Sustainability, Part III: Networking
- Infrastruttura globale di AWS
- Caratteristiche chiave di Amazon CloudFront, tra cui CloudFront Global Edge Network
- Compressing HTTP requests in Amazon OpenSearch Service
- Compressione dei dati intermedi con Amazon EMR
- Caricamento di file di dati compressi da Amazon S3 a Amazon Redshift
- Serving compressed files with Amazon CloudFront

Video correlati:

- Demystifying data transfer on AWS

SUS04-BP08 Backup dei dati solo quando sono difficili da ricreare

Evita il backup di dati senza valore aziendale per ridurre i requisiti delle risorse di archiviazione per il tuo carico di lavoro.

Anti-pattern comuni:

- Non hai una strategia di backup per i tuoi dati.
- Esegui il backup di dati che possono essere facilmente ricreati.

Vantaggi dell'adozione di questa best practice: riduzione delle risorse di archiviazione necessarie per il carico di lavoro, oltre al relativo impatto ambientale, evitando il backup di dati non critici.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Evitando il backup di dati non necessari si possono ridurre i costi e le risorse di archiviazione utilizzate dal carico di lavoro. Esegui il backup solo dei dati che hanno un valore aziendale o sono considerati necessari per soddisfare i requisiti di conformità. Esamina le policy di backup ed escludi l'archiviazione temporanea che non offre valore in uno scenario di ripristino.

Passaggi dell'implementazione

- Classifica i dati: implementa la policy di classificazione dei dati come illustrato in SUS04-BP01 Implementare una politica di classificazione dei dati.
- Progetta una strategia di backup: sfrutta la criticità della classificazione dei dati e progetta una strategia di backup basata su obiettivo del tempo di ripristino (RTO) e obiettivo del punto di ripristino (RPO). Evita il backup di dati non critici.
  - Escludi i dati che possono essere facilmente ricreati.
  - Escludi dati temporanei dai backup.
  - Escludi copie locali dei dati, a meno che il tempo necessario per ripristinare tali dati da una posizione comune superi gli accordi sul livello di servizio (SLA).
- Usa un backup automatico: usa una soluzione automatizzata o un servizio gestito per eseguire il backup di dati business-critical.
  - AWS Backup è un servizio totalmente gestito che semplifica la centralizzazione e l'automatizzazione della protezione dei dati in tutti i servizi AWS, nel cloud e on-premises. Per una guida pratica sulla creazione di backup automatici con AWS Backup, consulta Well-Architected Labs: test di backup e ripristino dei dati.
  - Automate backups and optimize backup costs for Amazon EFS using AWS Backup.

Risorse

Best practice correlate:

- REL09-BP01 Identificazione e backup di tutti i dati che richiedono un backup o riproduzione dei dati dalle origini
- REL09-BP03 Esecuzione del backup dei dati in automatico
- REL13-BP02 Utilizzo di strategie di ripristino definite per conseguire gli obiettivi di ripristino

Documenti correlati:

- Uso di AWS Backup per eseguire il backup e ripristinare i file system di Amazon EFS
- Amazon EBS snapshots
- Utilizzo dei backup su Amazon Relational Database Service
- Partner APN: partner per il backup
- Marketplace AWS: prodotti che possono essere utilizzati per il backup
- Backup Amazon EFS
- Backup di Amazon FSx per Windows File Server
- Backup e ripristino per Amazon ElastiCache (Redis OSS)

Video correlati:

- AWS re:Invent 2023 - Backup and disaster recovery strategies for increased resilience
- AWS re:Invent 2023 - What's new with AWS Backup
- AWS re:Invent 2021 - Backup, disaster recovery, and ransomware protection with AWS

Hardware e servizi

Domanda

- SUS 5. Come si selezionano e usano hardware e servizi cloud nell'architettura per supportare gli obiettivi di sostenibilità?

SUS 5. Come si selezionano e usano hardware e servizi cloud nell'architettura per supportare gli obiettivi di sostenibilità?

Cerca opportunità per ridurre l'impatto dei carichi di lavoro in termini di sostenibilità apportando modifiche alle tue pratiche di gestione hardware. Riduci al minimo la quantità di hardware necessaria per il provisioning e l'implementazione e scegli l'hardware e i servizi più efficienti per il singolo carico di lavoro.

Best practice

- SUS05-BP01 Utilizzo della quantità minima di hardware per soddisfare le esigenze aziendali
- SUS05-BP02 Utilizzo di tipi di istanze con il minimo impatto
- SUS05-BP03 Usa servizi gestiti

- SUS05-BP04 Ottimizzazione dell'uso degli acceleratori di calcolo basati su hardware

SUS05-BP01 Utilizzo della quantità minima di hardware per soddisfare le esigenze aziendali

Usa la quantità minima di hardware per il tuo carico di lavoro per soddisfare in modo efficiente le tue esigenze aziendali.

Anti-pattern comuni:

- Non monitori l'utilizzo delle risorse.
- Nella tua architettura sono presenti risorse con un basso livello di utilizzo.
- Non analizzi l'uso di hardware statico per stabilire se deve essere ridimensionato.
- Non imposti obiettivi di utilizzo dell'hardware per la tua infrastruttura di elaborazione in base a KPI aziendali.

Vantaggi dell'adozione di questa best practice: riduzione dell'impatto ambientale dei carichi di lavoro, risparmio di denaro e mantenimento dei benchmark delle prestazioni grazie al ridimensionamento corretto delle risorse cloud.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

Seleziona con precisione la quantità di hardware richiesta dal tuo carico di lavoro per migliorare l'efficienza generale. Cloud AWS offre la flessibilità necessaria per espandere o ridurre le risorse in modo dinamico attraverso una serie di meccanismi, ad esempio AWS Auto Scaling, in modo da soddisfare i cambiamenti della domanda. Offre anche API, SDK che consentono la modifica delle risorse con il minimo sforzo. Usa queste funzionalità per apportare modifiche frequenti alle implementazioni dei carichi di lavoro. Usa inoltre le linee guida sul ridimensionamento corretto degli strumenti AWS per gestire le risorse cloud in modo efficiente e soddisfare le esigenze aziendali.

Passaggi dell'implementazione

- Scegli il tipo di istanza: scegli il giusto tipo di istanza così da soddisfare appieno le tue esigenze. Per scoprire come scegliere le istanze Amazon Elastic Compute Cloud e utilizzare meccanismi quali la selezione delle istanze basata sugli attributi, consulta le seguenti risorse:
  - Come faccio a scegliere il tipo di istanza Amazon EC2 appropriata per il mio carico di lavoro?
  - Selezione del tipo di istanza basata su attributi per Amazon EC2 Fleet.

    - Crea un gruppo Auto Scaling utilizzando la selezione del tipo di istanza basata su attributi.
- Dimensiona usa piccoli incrementi per scalare carichi di lavoro variabili.
- Ricorri a più opzioni di acquisto di calcolo: bilancia flessibilità, scalabilità e risparmi sui costi delle istanze con più opzioni di acquisto di calcolo.
    - Le istanze on-demand di Amazon EC2 sono ideali per carichi di lavoro nuovi, stateful e con picchi, che non possono essere flessibili in termini di tipo di istanza, ubicazione o orario.
    - Le istanze spot di Amazon EC2 rappresentano una soluzione per l'integrazione di altre opzioni per applicazioni flessibili e tolleranti ai guasti.
    - Sfrutta i Savings Plans per il calcolo per carichi di lavoro a stato costante che garantiscono la flessibilità in caso di cambiamento delle le tue esigenze (come zone di disponibilità, regioni, famiglie di istanze o tipi di istanze).
- Usa la diversità di istanze e zone di disponibilità: ottimizza la disponibilità delle applicazioni e sfrutta la capacità in eccesso diversificando istanze e zone di disponibilità.
- Ridimensiona correttamente le istanze: segui le raccomandazioni per il dimensionamento di AWS per modificare il carico di lavoro. Per ulteriori informazioni, consulta Optimizing your cost with Rightsizing Recommendations e Right Sizing: Provisioning Instances to Match Workloads.
    - Puoi sfruttare i suggerimenti per il ridimensionamento corretto in AWS Cost Explorer o AWS Compute Optimizer per identificare le corrette opportunità di ridimensionamento corretto
- Negozia accordi sul livello di servizio (SLA): negozia SLA che consentano una riduzione temporanea della capacità quando l'automazione implementa risorse di sostituzione.

Risorse

Documenti correlati:

- Optimizing your AWS Infrastructure for Sustainability, Part I: Compute
- Attirbute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet
- Documentazione di AWS Compute Optimizer
- Operating Lambda: Performance optimization
- Documentazione su Auto Scaling

Video correlati:

- AWS re:Invent 2023 - What's new with Amazon EC2

- AWS re:Invent 2023 - Smart savings: Amazon Elastic Compute Cloud cost-optimization strategies
- AWS re:Invent 2022 - Optimizing Amazon Elastic Kubernetes Service for performance and cost on AWS
- AWS re:Invent 2023 - Sustainable compute: reducing costs and carbon emissions with AWS

SUS05-BP02 Utilizzo di tipi di istanze con il minimo impatto

Esegui un monitoraggio costante e usa nuovi tipi di istanza per sfruttare le migliorie in termini di efficienza energetica.

Anti-pattern comuni:

- Utilizzi una sola famiglia di istanze.
- Utilizzi solo istanze x86.
- Specifichi un tipo di istanza nella tua configurazione di Amazon EC2 Auto Scaling.
- Utilizzi istanze AWS in un modo per il quale non sono state progettate, ad esempio utilizzi istanze ottimizzate per il calcolo per un carico di lavoro a uso intensivo della memoria.
- Non valuti regolarmente l'uso di nuovi tipi di istanza.
- Non consulti i consigli ricevuti dagli strumenti di ridimensionamento corretto AWS, ad esempio AWS Compute Optimizer.

Vantaggi dell'adozione di questa best practice: l'uso di istanze energeticamente efficienti e di dimensioni corrette ti consente di ridurre in modo considerevole l'impatto ambientale e i costi del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

L'uso di istanze efficienti nel carico di lavoro cloud è fondamentale per ridurre l'utilizzo delle risorse e i costi. Monitora costantemente il rilascio di nuovi tipi di istanze e sfrutta le migliorie in tema di efficienza energetica, inclusi i tipi di istanze progettati per supportare carichi di lavoro specifici, come la formazione del machine learning, le inferenze e la transcodifica dei video.

Passaggi dell'implementazione

- Scopri e approfondisci i tipi di istanze: esplora e approfondisci i tipi di istanza in grado di ridurre l'impatto ambientale del carico di lavoro.

  - Abbonati a Novità di AWS per gli ultimi aggiornamenti in materia di istanze e tecnologie AWS.
  - Approfondisci i vari tipi di istanze AWS.
  - Scopri di più sulle istanze basate su AWS Graviton con le migliori prestazioni per watt di energia utilizzata in Amazon EC2 guardando re:Invent 2020 - Deep dive on AWS Graviton2 processor-powered Amazon EC2 instances e Deep dive into AWS Graviton3 and Amazon EC2 C7g instances.
- Usa i tipi di istanza che comportano il minor impatto: pianifica la transizione del carico di lavoro a tipi di istanza caratterizzati dal minimo impatto.
  - Definisci un processo per valutare nuove funzionalità o istanze per il carico di lavoro. Sfrutta l'agilità del cloud per testare in modo semplice e rapido in che modo i nuovi tipi di istanza possono migliorare la sostenibilità ambientale del carico di lavoro. Utilizza metriche proxy per misurare la quantità di risorse necessarie per completare un'unità di lavoro.
  - Se possibile, modifica il carico di lavoro in modo che funzioni con diversi numeri di CPU e quantità di memoria diverse per massimizzare la scelta del tipo di istanza.
  - Valuta l'ipotesi di trasferire il carico di lavoro in istanze basate su Graviton per migliorare l'efficienza delle prestazioni del carico di lavoro. Per ulteriori informazioni sullo spostamento dei carichi di lavoro su AWS Graviton, consulta Innova rapidamente con AWS Graviton Fast Start e Considerations when transitioning workloads to AWS Graviton-based Amazon Elastic Compute Cloud instances.
  - Valuta l'ipotesi di selezionare l'opzione AWS Graviton quando utilizzi i servizi gestiti AWS.
  - Esegui la migrazione del carico di lavoro nelle regioni che offrono istanze con il minor impatto in termini di sostenibilità e che contemporaneamente soddisfano i requisiti aziendali.
  - Per i carichi di lavoro di machine learning, sfrutta l'hardware specifico per il tuo carico di lavoro, come AWS Trainium, AWS Inferentia e Amazon EC2 DL1. AWS Le istanze Inferentia come le istanze Inf2 offrono fino al 50% in più di prestazioni per watt rispetto alle istanze Amazon EC2 paragonabili.
  - Usa Amazon SageMaker Inference Recommender per un endpoint di inferenza ML della giusta dimensione.
  - Per carichi di lavoro con picchi (carichi di lavoro con requisiti non frequenti di capacità aggiuntiva), utilizza istanze a prestazioni espandibili.
  - Per carichi di lavoro stateless e con tolleranza ai guasti, usa le istanze spot Amazon EC2 per aumentare l'utilizzo complessivo del cloud e ridurre l'impatto in termini di sostenibilità delle risorse inutilizzate.
- Esegui e ottimizza: esegui e ottimizza l'istanza del carico di lavoro.

- Per carichi di lavoro effimeri, valuta i parametri dell'istanza di Amazon CloudWatch, ad esempio `CPUUtilization`, per identificare se l'istanza è inattiva o sottoutilizzata.
- Per i carichi di lavoro stabili, esegui i controlli con gli strumenti di ridimensionamento corretto di AWS, come AWS Compute Optimizer, a intervalli regolari per individuare le opportunità di ottimizzazione e ridimensionamento corretto dell'istanza. Per ulteriori esempi e consigli, consulta i seguenti lab:
  - Well-Architected Lab: raccomandazioni per il ridimensionamento corretto
  - Well-Architected Lab: ridimensionamento corretto con Compute Optimizer
  - Well-Architected Lab: ottimizzazione dei modelli hardware e conformità agli indicatori KPI di sostenibilità

Risorse

Documenti correlati:

- Optimizing your AWS Infrastructure for Sustainability, Part I: Compute
- AWS Graviton
- Amazon EC2 DL1
- Amazon EC2 Capacity Reservation Fleets
- Amazon EC2 Spot Fleet
- Funzioni: configurazione della funzione Lambda
- Selezione del tipo di istanza basata su attributi per Amazon EC2 Fleet
- Building Sustainable, Efficient, and Cost-Optimized Applications on AWS
- How the Contino Sustainability Dashboard Helps Customers Optimize Their Carbon Footprint

Video correlati:

- AWS re:Invent 2023 - AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 - New Amazon Elastic Compute Cloud generative AI capabilities in Console di gestione AWS
- AWS re:Invent 2023 = What's new with Amazon Elastic Compute Cloud
- AWS re:Invent 2023 - Smart savings: Amazon Elastic Compute Cloud cost-optimization strategies
- AWS re:Invent 2021 - Deep dive into AWS Graviton3 and Amazon EC2 C7g instances

- AWS re:Invent 2022 - Build a cost-, energy-, and resource-efficient compute environment

Esempi correlati:

- Solution: Guidance for Optimizing Deep Learning Workloads for Sustainability on AWS

SUS05-BP03 Usa servizi gestiti

Usa i servizi gestiti per operare in modo più efficiente nel cloud.

Anti-pattern comuni:

- Utilizzi EC2 istanze Amazon a basso utilizzo per eseguire le tue applicazioni.
- Il tuo team interno gestisce solo il carico di lavoro, senza tempo per focalizzarsi sull'innovazione o sulle semplificazioni.
- Implementi e mantieni tecnologie per attività che possono essere eseguite in modo più efficiente sui servizi gestiti.

Vantaggi dell'adozione di questa best practice:

- L'uso dei servizi gestiti sposta la responsabilità verso AWS, che dispone di informazioni su milioni di clienti che possono contribuire a promuovere nuove innovazioni ed efficienze.
- Il servizio gestito distribuisce l'impatto ambientale del servizio su molti utenti a causa dei piani di controllo (control-plane) multi-tenet.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all'implementazione

I servizi gestiti trasferiscono la AWS responsabilità al mantenimento di un elevato utilizzo e all'ottimizzazione della sostenibilità dell'hardware distribuito. I servizi gestiti eliminano anche l'onere operativo e amministrativo legato alla manutenzione di un servizio, consentendo al tuo team di avere più tempo e di concentrarsi sull'innovazione.

Esamina il carico di lavoro per identificare i componenti che possono essere sostituiti dai AWS servizi gestiti. Ad esempio, Amazon RDS, Amazon Redshift e Amazon ElastiCache forniscono un servizio di database gestito. Amazon AthenaEMR, Amazon e Amazon OpenSearch Service forniscono un servizio di analisi gestito.

Passaggi dell'implementazione

1. Esegui l'inventario del carico di lavoro: esegui un inventario del tuo carico di lavoro in relazione a servizi e componenti.
2. Identifica i candidati: procedi a valutare e identificare i componenti sostituibili dai servizi gestiti. Ecco alcuni esempi in cui potresti prendere in considerazione l'uso di un servizio gestito:

Attività	Cosa usare su AWS
Ospitare un database	Utilizza istanze gestite di Amazon Relational Database Service (RDSAmazon) invece di mantenere le tue istanze Amazon su RDS Amazon Elastic Compute Cloud EC2 (Amazon).
Ospitare il carico di lavoro di un container	Utilizza AWS Fargate, invece di implementare un'infrastruttura di container proprietaria.
Ospitare applicazioni Web	Usa AWS Amplify Hosting come servizio CI/CD e di hosting completamente gestito per siti Web statici e app Web con rendering lato server.

3. Crea un piano di migrazione: individua le dipendenze e crea un piano di migrazione. Aggiorna runbook e playbook di conseguenza
   - AWS Application Discovery Service raccoglie e presenta automaticamente informazioni dettagliate sulle dipendenze e sull'utilizzo delle applicazioni per aiutarti a prendere decisioni più informate durante la pianificazione della migrazione.
4. Esegui i test: testa il servizio prima di migrare al servizio gestito.
5. Sostituisci i servizi in hosting autonomo: utilizza il tuo piano di migrazione per sostituire i servizi in hosting autonomo con servizi gestiti.
6. Monitora e modifica: monitora costantemente il servizio al termine della migrazione per apportare le modifiche richieste e ottimizzare il servizio.

Risorse

Documenti correlati:

- Cloud AWS Prodotti
- AWS Calcolatore del costo totale di proprietà (TCO)
- Amazon DocumentDB
- Servizio Amazon Elastic Kubernetes () EKS
- Streaming gestito da Amazon per Apache Kafka (Amazon) MSK

Video correlati:

- AWS re:Invent 2021 - Operazioni cloud su larga scala con AWS Managed Services
- AWS re:Invent 2023 - Le migliori pratiche per operare su AWS

SUS05-BP04 Ottimizzazione dell'uso degli acceleratori di calcolo basati su hardware

Ottimizza l'uso delle istanze a calcolo accelerato per ridurre i requisiti dell'infrastruttura fisica del carico di lavoro.

Anti-pattern comuni:

- Utilizzo delle GPU non monitorato.
- Utilizzo di un'istanza per uso generico per il carico di lavoro quando un'istanza appositamente sviluppata potrebbe offrire prestazioni più elevate, costi inferiori e migliori prestazioni per watt.
- Utilizzo di acceleratori di calcolo basati su hardware per attività in cui sono più efficienti le alternative basate su CPU.

Vantaggi dell'adozione di questa best practice: ottimizzando l'uso degli acceleratori basati su hardware, è possibile ridurre le richieste di infrastruttura fisica del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Se si necessita di un'elevata capacità di elaborazione, si può trarre vantaggio dall'uso di istanze a calcolo accelerato, che forniscono l'accesso ad acceleratori di calcolo basati su hardware, come le unità di elaborazione grafica (GPU) e le serie di porte programmabili sul campo (FPGA) Questi acceleratori hardware eseguono alcune funzioni, come l'elaborazione grafica o la rilevazione della corrispondenza dei modelli di dati, in modo più efficiente rispetto alle alternative basate su CPU. Molti

carichi di lavoro accelerati, come il rendering grafico, la transcodifica e il machine learning, sono altamente variabili in termini di utilizzo di risorse. Mantieni in esecuzione questo tipo di hardware solo per il tempo necessario e disattivalo automaticamente quando non serve per ridurre la quantità di risorse utilizzate.

Passaggi dell'implementazione

- Acceleratori di calcolo: identifica le istanze a calcolo accelerato in grado di soddisfare i tuoi requisiti.
- Utilizzo di hardware appositamente progettato: per i carichi di lavoro di machine learning, sfrutta l'hardware specifico per il tuo carico di lavoro, come AWS Trainium, AWS Inferentia e Amazon EC2 DL1. AWS Le istanze Inferentia come le istanze Inf2 offrono fino al 50% in più di prestazioni per watt rispetto alle istanze Amazon EC2 paragonabili.
- Monitoraggio delle metriche di utilizzo: raccogli le metriche di utilizzo per le tue istanze a calcolo accelerato. Ad esempio, puoi utilizzare l'agente CloudWatch per acquisire metriche quali `utilization_gpu` e `utilization_memory` per le tue GPU, come illustrato in Collect NVIDIA GPU metrics with Amazon CloudWatch.
- Dimensionamento corretto: ottimizza il codice, il funzionamento della rete e le impostazioni degli acceleratori hardware per garantire il pieno utilizzo dell'hardware sottostante.
  - Ottimizza le impostazioni GPU
  - Monitoraggio e ottimizzazione delle GPU nell'AMI per il deep learning
  - Optimizing I/O for GPU performance tuning of deep learning training in Amazon SageMaker
- Sempre al passo: utilizza le librerie e i driver per GPU più recenti e performanti.
- Rilascio di istanze non necessarie: utilizza l'automazione per rilasciare le istanze GPU non in uso.

Risorse

Documenti correlati:

- Calcolo accelerato
- Let's Architect! Architecting with custom chips and accelerators
- Come faccio a scegliere il tipo di istanza Amazon EC2 appropriata per il mio carico di lavoro?
- Amazon EC2 VT1 Instances
- Choose the best AI accelerator and model compilation for computer vision inference with Amazon SageMaker

Video correlati:

- AWS re:Invent 2021 - How to select Amazon EC2 GPU instances for deep learning
- AWS Online Tech Talks - Deploying Cost-Effective Deep Learning Inference
- AWS re:Invent 2023 - Cutting-edge AI with AWS and NVIDIA
- AWS re:Invent 2022 - [NEW LAUNCH!] Introducing AWS Inferentia2-based Amazon EC2 Inf2 instances
- AWS re:Invent 2022 - Accelerate deep learning and innovate faster with AWS Trainium
- AWS re:Invent 2022 - Deep learning on AWS with NVIDIA: From training to deployment

Processo e cultura

Domanda

- SUS 6. In che modo i processi organizzativi possono supportare gli obiettivi di sostenibilità?

SUS 6. In che modo i processi organizzativi possono supportare gli obiettivi di sostenibilità?

Cerca opportunità per ridurre l'impatto di sostenibilità apportando modifiche alle tue prassi di sviluppo, test e implementazione.

Best practice

- SUS06-BP01 Comunicazione e collaborazione per gli obiettivi di sostenibilità
- SUS06-BP02 Adozione di metodi che consentano di introdurre rapidamente migliorie in tema di sostenibilità
- SUS06-BP03 Aggiornamento del carico di lavoro
- SUS06-BP04 Incremento dell'utilizzo degli ambienti di compilazione
- SUS06-BP05 Utilizzo di device farm gestite per i test

SUS06-BP01 Comunicazione e collaborazione per gli obiettivi di sostenibilità

La tecnologia è un fattore chiave per la sostenibilità. I team IT svolgono un ruolo cruciale nel promuovere cambiamenti significativi per il raggiungimento degli obiettivi di sostenibilità dell'organizzazione. Questi team devono comprendere chiaramente gli obiettivi di sostenibilità

dell'azienda e lavorare per comunicare tali priorità e integrarle in modo collaborativo tra le varie attività.

Anti-pattern comuni:

- Non conosci gli obiettivi di sostenibilità della tua organizzazione e come si applicano al tuo team.
- Hai una consapevolezza e una formazione insufficienti sull'impatto ambientale dei carichi di lavoro cloud.
- Non sai quali sono le aree specifiche a cui dare priorità.
- Non coinvolgi dipendenti e clienti nelle iniziative di sostenibilità.

Vantaggi derivanti dall'adozione di questa best practice: dall'ottimizzazione dell'infrastruttura e dei sistemi all'uso di tecnologie innovative, i team IT possono abbassare le emissioni di carbonio dell'organizzazione e ridurre al minimo l'utilizzo delle risorse. La comunicazione degli obiettivi di sostenibilità può offrire ai team IT la possibilità di migliorare e adattarsi continuamente alle mutevoli problematiche legate alla sostenibilità. Queste ottimizzazioni sostenibili spesso si traducono anche in risparmi sui costi, con conseguente rafforzamento del business case.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

I principali obiettivi di sostenibilità per i team IT dovrebbero essere l'ottimizzazione di sistemi e soluzioni per aumentare l'efficienza delle risorse e ridurre al minimo l'impronta di carbonio e l'impatto ambientale complessivo dell'organizzazione. Servizi e iniziative condivisi, come programmi di formazione e dashboard operative, possono supportare le organizzazioni nell'ottimizzazione delle operazioni IT e nella creazione di soluzioni che contribuiscono a ridurre in modo significativo l'impronta di carbonio. Il cloud offre l'opportunità non solo di trasferire le responsabilità dell'infrastruttura fisica e dell'approvvigionamento energetico alla responsabilità condivisa del fornitore di servizi cloud, ma anche di ottimizzare continuamente l'efficienza delle risorse dei servizi basati sul cloud.

Quando i team utilizzano l'efficienza intrinseca e il modello di responsabilità condivisa del cloud, possono ottenere riduzioni significative dell'impatto ambientale dell'organizzazione. Questo, a sua volta, aiuta a raggiungere gli obiettivi complessivi di sostenibilità dell'organizzazione e a dimostrare il valore di questi team come partner strategici nel percorso verso un futuro più sostenibile.

Passaggi dell'implementazione

- Definisci traguardi e obiettivi: stabilisci obiettivi ben definiti per il tuo programma IT. Ciò implica ricevere input dalle parti interessate responsabili di diversi dipartimenti, come quelli che si occupano di IT, sostenibilità e finanza. Questi team devono definire obiettivi misurabili che siano in linea con gli obiettivi di sostenibilità dell'organizzazione, comprese aree come la riduzione delle emissioni di carbonio e l'ottimizzazione delle risorse.
- Comprendi i limiti correlati alla contabilità del carbonio della tua azienda: scopri in che modo i metodi per la contabilità del carbonio, come il protocollo Greenhouse Gas (GHG), si relazionano ai tuoi carichi di lavoro nel cloud (per maggiori dettagli, consulta Sostenibilità del cloud).
- Utilizza soluzioni cloud per la contabilità del carbonio: utilizza soluzioni cloud come le soluzioni per la contabilità del carbonio di AWS per tenere traccia degli scope 1, 2 e 3 per le emissioni di gas a effetto serra nelle tue attività, nei tuoi portfolio e nelle tue catene del valore. Con queste soluzioni, le organizzazioni possono semplificare l'acquisizione dei dati sulle emissioni di gas a effetto serra, semplificare la creazione di report e ricavare approfondimenti utili per le proprie strategie climatiche.
- Monitora l'impronta di carbonio del tuo portfolio IT: monitora le emissioni di carbonio dei tuoi sistemi IT e crea report con i relativi dati. Utilizza il AWS Customer Carbon Footprint Tool per monitora, misurare, esaminare e prevedere le emissioni di carbonio generate dall'utilizzo del tuo ambiente AWS.
- Comunica ai tuoi team l'utilizzo delle risorse tramite metriche proxy: monitora l'utilizzo delle risorse tramite metriche proxy e crea report con i relativi dati. Nei modelli di prezzo on demand del cloud, l'utilizzo delle risorse è correlato ai costi, che rappresentano una metrica comprensibile a livello generale. Utilizza i costi come metrica proxy almeno per comunicare l'utilizzo delle risorse e i miglioramenti da parte di ciascun team.
  - Abilita la granularità oraria nell'Esploratore dei costi e crea un report di costi e utilizzo (CUR): il report CUR offre granularità di utilizzo, tariffe, costi e attributi di utilizzo su base oraria o giornaliera per tutti i servizi AWS. Utilizza Cloud Intelligence Dashboards e la relativa Sustainability Proxy Metrics Dashboard come punto di partenza per l'elaborazione e la visualizzazione dei dati in base a costi e utilizzo. Per ulteriori dettagli, consulta i seguenti riferimenti:
  - Measure and track cloud efficiency with sustainability proxy metrics, Part I: What are proxy metrics?
  - Measure and track cloud efficiency with sustainability proxy metrics, Part II: Establish a metrics pipeline

- Ottimizza e valuta in modo continuo: utilizza un processo di miglioramento per ottimizzare continuamente i tuoi sistemi IT, incluso il carico di lavoro cloud per l'efficienza e la sostenibilità. Monitora l'impronta di carbonio prima e dopo l'implementazione della strategia di ottimizzazione. Utilizza la riduzione dell'impronta di carbonio per valutarne l'efficacia.
- Promuovi una cultura della sostenibilità: utilizza programmi di formazione (come AWS Skill Builder) per educare i dipendenti alla sostenibilità. Coinvolgi i dipendenti in iniziative legate alla sostenibilità. Condividi e celebra le loro storie di successo. Utilizza gli incentivi per offrire premi in caso di raggiungimento degli obiettivi di sostenibilità.

Risorse

Documenti correlati:

- Understanding your carbon emission estimations

Video correlati:

- AWS re:Invent 2023 - Accelerate data-driven circular economy initiatives with AWS
- AWS re:Invent 2023 - Sustainability innovation in AWS Global Infrastructure
- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2022 - Architecting sustainably and reducing your AWS carbon footprint
- AWS re:Invent 2022 - Sustainability in AWS global infrastructure

Esempi correlati:

- Well-Architected Lab: trasformare i report su costi e utilizzo in report sull'efficienza

Formazione correlata:

- Sustainability Transformation on AWS
- SimuLearn - Sustainability Reporting
- Decarbonization with AWS

SUS06-BP02 Adozione di metodi che consentano di introdurre rapidamente migliorie in tema di sostenibilità

Adotta metodi e processi per convalidare migliorie potenziali, ridurre i costi legati ai test e offrire piccole migliorie.

Anti-pattern comuni:

- Analizzare l'applicazione rispetto alla sostenibilità è un'attività che viene eseguita solo una volta, all'inizio di un progetto.
- Il tuo carico di lavoro non è aggiornato, poiché il processo di rilascio è troppo complesso per introdurre modifiche minori per l'efficienza delle risorse.
- Non hai meccanismi per migliorare il tuo carico di lavoro in termini di sostenibilità.

Vantaggi dell'adozione di questa best practice: la definizione di un processo per l'introduzione e il monitoraggio dei miglioramenti della sostenibilità consente di adottare in modo continuo nuove funzionalità e funzioni, risolvere i problemi e migliorare l'efficienza del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: medio

Guida all’implementazione

Testa e convalida potenziali miglioramenti all'impatto sulla sostenibilità prima di implementarli in produzione. Tieni in considerazione il costo dei test quando calcoli il potenziale vantaggio futuro di un miglioramento. Sviluppa metodi di test a basso costo per consentire la distribuzione di piccoli miglioramenti.

Passaggi dell’implementazione

- Analizza e comunica i tuoi obiettivi di sostenibilità organizzativa: esamina i tuoi obiettivi di sostenibilità organizzativa, come la riduzione delle emissioni di carbonio o la gestione delle risorse idriche. Traduci questi obiettivi in requisiti di sostenibilità per i carichi di lavoro del cloud. Comunica questi requisiti alle principali parti interessate.
- Aggiungi i requisiti di sostenibilità al tuo backlog: aggiungi i requisiti relativi al miglioramento della sostenibilità al tuo backlog di sviluppo.
- Itera e migliora: utilizza un processo di miglioramento iterativo per identificare, valutare, assegnare priorità, testare e implementare questi miglioramenti.
- Esegui test utilizzando il prodotto minimo funzionante (MVP): sviluppa e testa potenziali miglioramenti con componenti minimi funzionanti per ridurre costi e impatto ambientale dei test.

- Semplifica il processo: migliora e semplifica continuamente i tuoi processi di sviluppo. Ad esempio, automatizza il processo di distribuzione del software con pipeline di distribuzione e integrazione continue (CI/CD) per testare e implementare migliorie potenziali per ridurre il livello di impegno e gli errori causati da processi manuali.
- Gestisci formazione e sensibilizzazione: organizza programmi di formazione per i membri del tuo team per sensibilizzarli in merito alla sostenibilità e sull'impatto delle loro attività sugli obiettivi di sostenibilità dell'organizzazione.
- Valuta e modifica: valuta in modo costante l'impatto delle migliorie e apporta gli adeguamenti richiesti.

Risorse

Documenti correlati:

- AWS consente soluzioni di sostenibilità

Video correlati:

- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2022 - Architecting sustainably and reducing your AWS carbon footprint
- AWS re:Invent 2022 - Sustainability in AWS global infrastructure
- AWS re:Invent 2023 - What's new with AWS observability and operations

SUS06-BP03 Aggiornamento del carico di lavoro

Aggiorna il tuo carico di lavoro per adottare funzionalità efficienti, eliminare le problematiche e migliorare l'efficienza generale del tuo carico di lavoro.

Anti-pattern comuni:

- Si ritiene che l'architettura corrente diventi statica e non venga aggiornata nel corso del tempo.
- Non si dispone di sistemi né si esegue regolarmente una valutazione per la compatibilità di software e pacchetti aggiornati con il carico di lavoro.

Vantaggi dell'adozione di questa best practice: la definizione di un processo per garantire il costante aggiornamento del carico di lavoro ti consentirà di adottare nuove caratteristiche e funzionalità, risolvere i problemi e migliorare l'efficienza del carico di lavoro.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Sistemi operativi, runtime, middleware (software intermediario), librerie e applicazioni aggiornati possono incidere sull'efficienza dei carichi di lavoro e facilitano l'adozione delle tecnologie più efficienti. Il software aggiornato potrebbe anche includere funzionalità per misurare in modo più accurato l'impatto in termini di sostenibilità del carico di lavoro, poiché i fornitori offrono caratteristiche per raggiungere i propri obiettivi di sostenibilità. Adotta una cadenza regolare per aggiornare il tuo carico di lavoro con le ultime funzionalità e i rilasci più recenti.

Passaggi dell'implementazione

- Definisci un processo: serviti di un processo e una pianificazione per valutare nuove funzionalità o istanze per il carico di lavoro. Sfrutta l'agilità del cloud per testare in modo semplice e rapido il modo in cui le nuove funzionalità possono migliorare il carico di lavoro nei seguenti ambiti:
  - Riduzione dell'impatto a livello di sostenibilità.
  - Raggiungimento di maggiore efficienza in termini di prestazioni.
  - Eliminazione delle barriere finalizzata a un miglioramento pianificato.
  - Miglioramento della capacità di misurare e gestire l'impatto a livello di sostenibilità.
- Esegui l'inventario: redigi l'inventario del software e dell'architettura del carico di lavoro e identifica i componenti che richiedono un aggiornamento.
  - Puoi usare l'inventario di AWS Systems Manager per raccogliere i metadati relativi a sistema operativo, applicazioni e istanze dalle istanze Amazon EC2 per una panoramica immediata su quali istanze stanno eseguendo il software e le configurazioni richieste dalle policy software e quali istanze vanno aggiornate.
- Apprendi la procedura di aggiornamento: scopri come aggiornare i componenti del carico di lavoro.

Componente del carico di lavoro	Come aggiornare
Immagini della macchina	Usa EC2 Image Builder per gestire gli aggiornamenti Amazon Machine Image (AMI) per Linux o Windows.
Immagini di container	Usa Amazon Elastic Container Registry (Amazon ECR) con la tua pipeline esistente per gestire le immagini di Amazon Elastic Container Service (Amazon ECS).
AWS Lambda	AWS Lambda include funzionalità di gestione delle versioni.

- Utilizza l'automazione: usa l'automazione degli aggiornamenti per ridurre il livello di impegno per implementare le nuove funzionalità e limitare gli errori causati dai processi manuali.
  - Puoi usare CI/CD per aggiornare in automatico AMI, immagini di container e altri artefatti relativi alla tua applicazione cloud.
  - È possibile utilizzare strumenti come Gestione patch di AWS Systems Manager per automatizzare il processo di aggiornamento del sistema e pianificare l'attività utilizzando le Finestre di manutenzione di AWS Systems Manager.

Risorse

Documenti correlati:

- AWS Architecture Center
- Novità di AWS
- Strumenti per sviluppatori in AWS

Video correlati:

- AWS re:Invent 2022 - Optimize your AWS workloads with best-practice guidance
- All Things Patch: AWS Systems Manager

SUS06-BP04 Incremento dell'utilizzo degli ambienti di compilazione

Aumenta l'uso delle risorse per sviluppare, testare e creare i tuoi carichi di lavoro.

Anti-pattern comuni:

- Esegui il provisioning manuale o interrompi i tuoi ambienti di sviluppo.
- Fai in modo che i tuoi ambienti di sviluppo siano in esecuzione indipendentemente dalle attività di test, creazione o rilascio (ad esempio, eseguire un ambiente al di fuori dell'orario di lavoro dei membri del tuo team di sviluppo).
- Esegui un provisioning eccessivo delle tue risorse per gli ambienti di creazione.

Vantaggi dell'adozione di questa best practice: l'aumento dell'utilizzo degli ambienti di compilazione migliora l'efficienza complessiva del carico di lavoro in cloud, allocando al contempo le risorse agli sviluppatori per sviluppo, test e compilazione ottimali.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

Utilizza automazione e modelli Infrastructure as code per rendere operativi gli ambienti di produzione quando necessario e dismetterli quando non vengono utilizzati. Un modello comune consiste nel pianificare periodi di disponibilità che coincidano con l'orario di lavoro dei membri del team incaricati dello sviluppo. Gli ambienti di test devono essere molto simili alla configurazione di produzione. Tuttavia, cerca la possibilità di utilizzare tipi di istanze con capacità di espansione, istanze spot Amazon EC2, servizi di database con dimensionamento automatico, container e tecnologie serverless per allineare la capacità di sviluppo e test all'uso. Limita i volumi di dati per soddisfare solo i requisiti di test. Se usi i dati di produzione per i test, rifletti sulla possibilità di condividere i dati di produzione invece di spostarli.

Passaggi dell'implementazione

- Utilizza il modello Infrastructure as code: usa il modello Infrastructure as code per eseguire il provisioning dei tuoi ambienti di sviluppo.
- Utilizza l'automazione: usa l'automazione per gestire il ciclo di vita degli ambienti di sviluppo e test e massimizzare l'efficienza delle tue risorse di sviluppo.
- Massimizza l'utilizzo: utilizza strategie per ottimizzare l'utilizzo degli ambienti di sviluppo e test.
  - Utilizza ambienti rappresentativi minimi realizzabili per lo sviluppo e il test di potenziali miglioramenti.

- Utilizza tecnologie serverless, se possibile.
- Utilizza istanze on-demand per integrare i dispositivi per gli sviluppatori.
- Utilizza i tipi di istanze con capacità di espansione, istanze spot e altre tecnologie per allineare la capacità di compilazione all'uso.
- Adotta servizi cloud nativi per un accesso sicuro agli shell (interprete di comandi) delle istanze invece di implementare parchi istanze di host bastioni.
- Dimensiona automaticamente le tue risorse di sviluppo in base alle tue attività.

Risorse

Documenti correlati:

- AWS Systems Manager Session Manager
- Istanze a prestazioni espandibili di Amazon EC2
- Che cos'è AWS CloudFormation?
- Che cos'è AWS CodeBuild?
- Pianificatore di istanze su AWS

Video correlati:

- AWS re:Invent 2023 - Continuous integration and delivery for AWS

SUS06-BP05 Utilizzo di device farm gestite per i test

Usa device farm gestite per testare in maniera efficiente una nuova funzionalità su un set rappresentativo di hardware.

Anti-pattern comuni:

- Testa e implementi manualmente la tua applicazione su singoli dispositivi fisici.
- Non utilizzi il servizio di test delle app per testare e interagire con le tue app (ad esempio, Android, iOS e app Web) su dispositivi fisici reali.

Vantaggi dell'adozione di questa best practice: l'utilizzo di farm di dispositivi gestiti per il test delle applicazioni abilitate al cloud offre una serie di vantaggi.

- Offrono funzionalità più efficienti per testare le applicazioni su un'ampia gamma di dispositivi.
- Eliminano la necessità di un'infrastruttura in-house per i test.
- Offrono diverse tipologie di dispositivi, tra cui hardware di generazioni precedenti e meno diffuso, eliminando così la necessità di aggiornamenti non necessari dei dispositivi.

Livello di rischio associato se questa best practice non fosse adottata: basso

Guida all'implementazione

L'uso di device farm gestite può aiutarti a semplificare il processo di test per le nuove funzionalità su un gruppo rappresentativo di hardware. Le device farm gestite offrono diversi tipi di dispositivi, inclusi hardware meno diffusi e di generazioni precedenti, ed evitano l'impatto sulla sostenibilità dei clienti dovuti ad aggiornamenti dei dispositivi non necessari.

Passaggi dell'implementazione

- Definisci i requisiti di test: definisci i requisiti di test ed esegui la pianificazione (come tipo di test, sistemi operativi e programma di test).
  - Amazon CloudWatch RUM ti consente di raccogliere e analizzare i dati lato client e formulare il tuo piano di test.
- Seleziona una device farm gestita: scegli una device farm gestita in grado di supportare i tuoi requisiti di test. Ad esempio, puoi utilizzare AWS Device Farm per testare e analizzare l'impatto delle modifiche su un set di hardware rappresentativo.
- Utilizza l'automazione: usa automazione e integrazione continua/l'implementazione continua (CI/CD) per pianificare ed eseguire i test.
  - Integrating AWS Device Farm with your CI/CD pipeline to run cross-browser Selenium tests
  - Building and testing iOS and iPadOS apps with AWS DevOps and mobile services
- Rivedi e modifica: esamina sempre i risultati dei test e apporta le migliorie richieste.

Risorse

Documenti correlati:

- Elenco dei dispositivi AWS Device Farm
- Viewing the CloudWatch RUM dashboard

Video correlati:

- AWS re:Invent 2023 - Improve your mobile and web app quality using AWS Device Farm
- AWS re:Invent 2021 - Optimize applications through end user insights with Amazon CloudWatch RUM

Esempi correlati:

- App di esempio AWS Device Farm per Android
- App di esempio AWS Device Farm per iOS
- Test Appium Web per AWS Device Farm

Note

I clienti sono responsabili della propria valutazione indipendente delle informazioni contenute nel presente documento. Questo documento: (a) è esclusivamente a scopo informativo, (b) rappresenta le attuali offerte e pratiche dei prodotti AWS, soggette a modifiche senza preavviso, e (c) non costituisce alcun impegno o garanzia da parte di AWS e dei suoi affiliati, fornitori o licenziatari. I prodotti o i servizi AWS sono forniti nello stato di fatto in cui si trovano, senza garanzie, dichiarazioni o condizioni di alcun tipo, sia esplicite che implicite. Le responsabilità di AWS nei confronti dei propri clienti sono definite dai contratti AWS e il presente documento non costituisce parte né modifica qualsivoglia contratto tra AWS e i suoi clienti.

AWS Glossario

Per la AWS terminologia più recente, consultate il AWS glossario nella sezione Reference.Glossario AWS